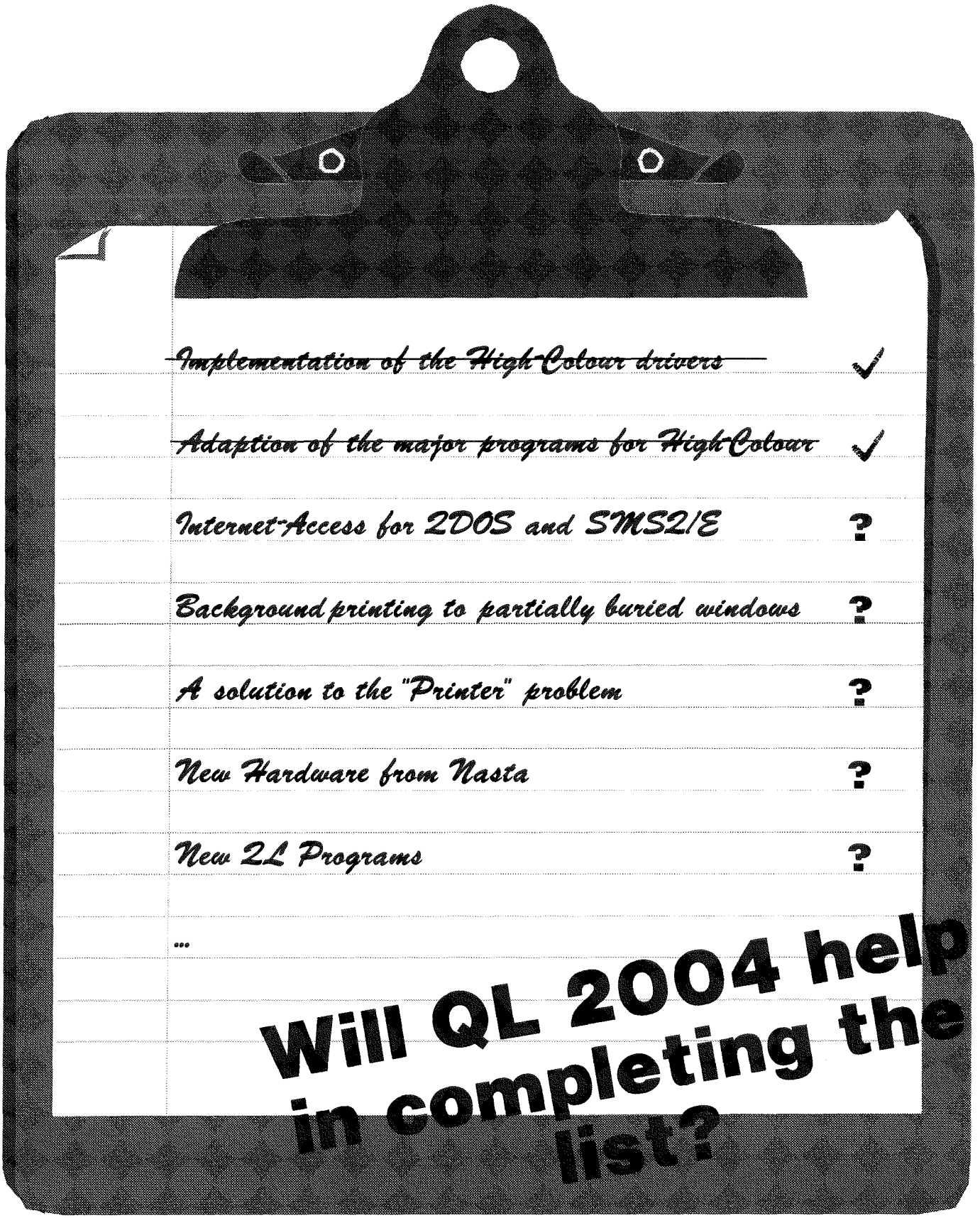
QL Today

Volume 9
Issue 3
August-Oct.
2004

ISSN 1432-5454

The Magazine about QL, QDOS, Sinclair Computers, SMSQ...



Contents

Advertisers

in alphabetical order

QL Today

ISSN 1432-5454

German office & Publisher:

Jochen Merz Software
Kaiser-Wilhelm-Str. 302
47169 Duisburg
Germany

Tel. +49 203 502011
Fax +49 203 502012
email: JMerz@j-m-s.com
email: QLToday@j-m-s.com

English office:

Q Branch
20 Locks Hill
Portslade
BN41 2LB
United Kingdom

Tel. +44 1273 386030
Mobile +44 7836 745501
Fax +44 1273 381577
email: qbranch@qbranch.demon.co.uk
email: QLToday@j-m-s.com

Editor:

Dilwyn Jones
41 Bro Emrys
Tal-Y-Bont, Bangor
Gwynedd
United Kingdom LL57 3YT

email: dilwyn.jones@tesco.net
email: QLToday@J-M-S.com

Co-Editor:

Bruce Nicholls
38 Derham Gardens
Upminster
Essex RM14 3HA
United Kingdom

Tel +44 1708 510764
Fax +44 870 0568755
email: qltoday@q-v-d.demon.co.uk
email: QLToday@j-m-s.com

QL Today is published bi-monthly, our volume begins on beginning of June. Subscriptions begin with the current issue at the time of sign up. Please contact the German or English office for current subscription rates.

We welcome your comments, suggestions and articles. YOU make **QL Today** possible. We are constantly changing and adjusting to meet your needs and requirements. Articles for publication should be on a 3.5" disk (DD or HD) or sent via Email. We prefer ASCII, Quill or text87 format. Pictures may be in _SCR format, we can also handle GIF or TIF or JPG. To enhance your article you may wish to include Saved Screen dumps. PLEASE send a hardcopy of all screens to be included. Don't forget to specify where in the text you would like the screen placed.

The deadline for Issue 4 is the 20th of November!

Editorial

Dilwyn Jones

A bit of the old and a bit of the new this month.

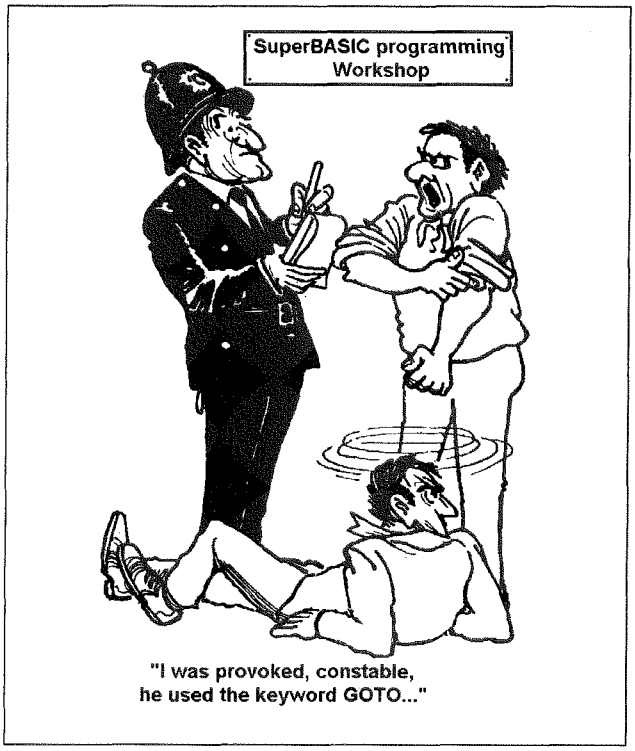
With QL2004 in Eindhoven nearly upon us as I write this, we take a look back to the early production of QLs back in the 1980s, as Simon Goodwin sets out the question of "just how many QLs were made?" The article is also interesting reading given that we are in the 20th year of QLing and although I thought I knew just about everything there was to know about QLs, I learned a lot from reading this article.

Although all those thousands of QLs were made, very few are to be seen at QL shows these days. Just about all machines now are re-cased QLs, Q40/Q60s or emulators. It probably says something about how we have moved on. Although we mustn't assume everyone is into the latest hardware and software – there are still users out there using Trump Cards and Gold Cards who've never tried SMSQ/E or even got used to pointer environment and we mustn't lose sight of them!

An important aspect of QLing these days is the amount of free software available for QDOS and SMSQ/E systems. I've been gradually putting my PD library online and it's proved to be a massive job, there's so much software out there. John Perry sent me a little article to try to explain the various kinds of soft-wares out there, and that inspired me to take a good look at what's available out there on the World Wide Web. For a computer which has not yet got a full online capability, there's an awful lot of sites out there with QL related material on them, and of course a lot of them have software to download, most of it free of charge. In fact, I was pleasantly surprised when I went to research just what was out there. If you have access to the internet, try the websites of Thierry Godefroy, Giorgio Garabello, Jonathan Hudson and Miguel Angel Hernandez as some examples of what's available. Phoebus Dokos has applied to produce a database driven software download site as part of his graduation project, which should make it easier for software writers to upload and make available their work, not to mention making it easier for users to search for what they want!

It's been a quiet period as far as news is concerned. I hope that it's because it's been summer and that any news is aimed for the QL2004 event in mid October. Traditionally, the autumn/fall period is when activity picks up (sell ice cream in the summer and computer software in the winter and all that). Now that reminds me, what do I want for my QL or QPC for Christmas? Santa Jochen will need to know soon...

Cartoon



"I was provoked, constable, he used the keyword GOTO..."

NEWS

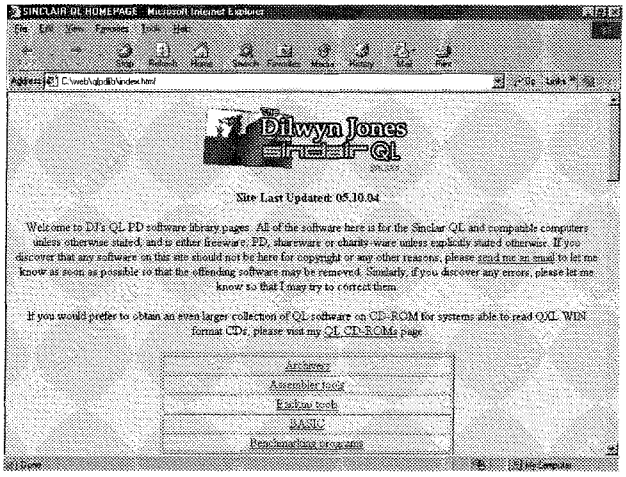
Software Download Site

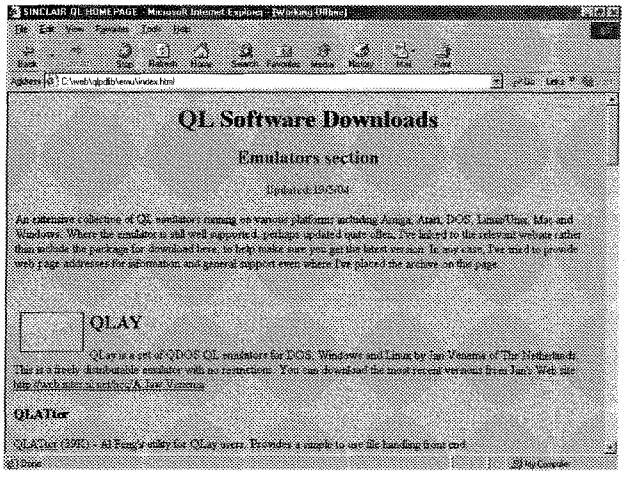
My software download site has been moved on a trial basis to:

www.dilwyn.uk6.net/index.htm

You can also reach it via the link from my main tesco.net website's home page.

The content is largely the same as it was when on **www.dokos-gr.net**. There's still a few 'under construction' pages which I'll finish as and when I get time. It took ages to upload the 50MB onto the new site even though I did it via ISDN at work rather than the 56k dialup from home. I have already started to add material there as I get time. The old site used to have just under 50MB, at the time of writing this there's 66MB and plenty more material to add, and plenty of space available.





A couple of pages from the new website

QL-Archive Site

Phoebus Dokos has embarked on an ambitious project as part of his graduation work for this year. It's destined to be a database driven software archive site for the QL, allowing you to both download and upload software to it. Probably a bit like you used to be able to do with some bulletin board systems, for example. With most sites currently (like mine) you have to send an email to the owner, asking if they'd like to receive a file, email it to them, and wait for the emailed software to be be accepted and placed on the website. Hopefully, this will take QL software availability a step further! Here is the email which Phoebus sent about the project (slightly edited):

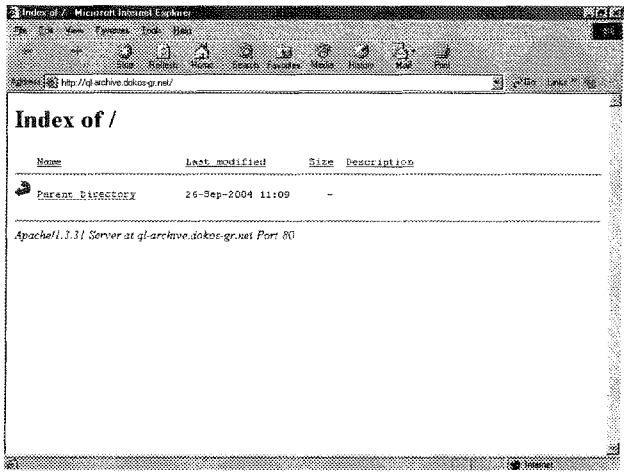
"For the record the address is

http://ql-archive.dokos-gr.net

The good news is that the database structure is already done. The better news is that this has been accepted as my graduating project for my MIS Database courses.

The super cool news is that you will be able to upload stuff yourself on it.

For the time being I am converting archives for use with that (i.e. all my backups from various archives are being tabulated and converted in the appropriate format for upload to the database (i.e. conversion to comma-delimited format text files to be uploaded to mySQL)."



At the time of writing, the page existed but no software yet available from it

Just Words Software Updates

Geoff Wicks writes:

Although the official release of the Just Words! GD2 upgrades is programmed for next weekend at QL2004, they are now available for downloading from my website. I have done this prerelease to take pressure off me at the show, because I shall have little time for trading there.

There are two ways of downloading the upgrades:

1: If you possess a lot of Just Words! freeware then go to the file "GD2 freeware upgrades" near the top of the page. This contains the _obj files only of:

SOLVIT-PLUS
QL-THESAURUS
NL-THESAURUS (DUTCH)
STYLE-CHECK
STIJL-CHECK (DUTCH)
SPELLING CRIB

These programs will work with your existing configuration in existing _def files, although you may need to reconfigure the colours in SOLVIT-PLUS, STYLE-CHECK and STIJL-CHECK to get the display you want.

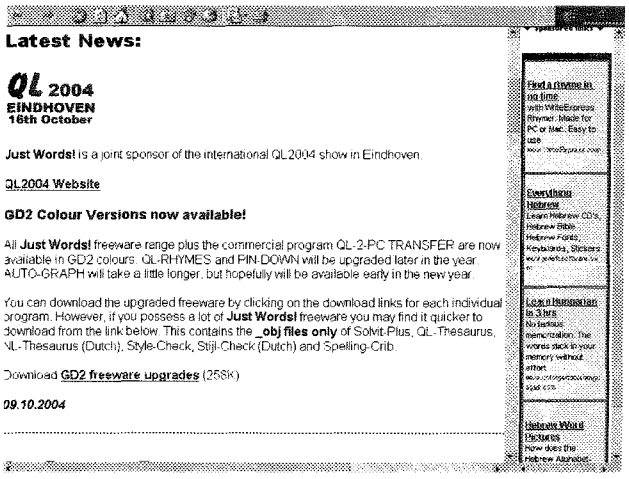
2: If you only have one or two Just Words! freeware programs you can download the individual program files as these also contain the GD2 versions.

New on the site are downloadable versions of NL-THESAURUS, STIJL-CHECK and the USA spellings database for QL-THESAURUS.

A GD2 version of QL-2-PC TRANSFER will be available at QL2004. As this is a commercial program you will have to produce your master disc on which the new version will be saved. You must have the menu_ext extensions installed on your machine to use the GD2 version of this program.

I hope to upgrade QL-RHYMES and PIN-DOWN to GD2 versions before the end of the year. AUTO-GRAPH will take a little longer because of the large number of sprites in the program. Hopefully this will be available early in the new year.

http://members.lycos.co.uk/geoffwicks/justwords.htm



Latest News:

QL 2004
EINDHOVEN
16th October

Just Words! is a joint sponsor of the international QL2004 show in Eindhoven.

QL2004 Website

GD2 Colour Versions now available!

All **Just Words!** freeware range plus the commercial program QL-2-PC TRANSFER are now available in GD2 colours. QL-RHYMES and PIN-DOWN will be upgraded later in the year. AUTO-GRAPH will take a little longer, but hopefully will be available early in the new year.

You can download the upgraded freeware by clicking on the download links for each individual program. However, if you possess a lot of **Just Words!** freeware you may find it quicker to download from the link below. This contains the **_obj files only** of Solvit-Plus, QL-Thesaurus, NL-Thesaurus (Dutch), Style-Check, Stijl-Check (Dutch) and Spelling-Crib.

Download GD2 freeware upgrades (256K)

09.10.2004

Geoff's announcement on his website

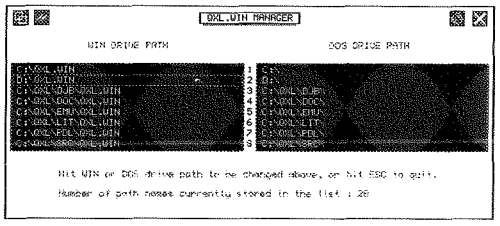
QXL.WIN Manager

This is a new pointer driven program for QPC2 users to help with setting the path names for the 8 WIN and DOS devices available. If, like me, you are in the habit of chopping and changing about the names used for the WIN and DOS devices and have problems remembering all the path names for the various locations, help is at hand. This program remembers the path names you have already used and maintains a list of these. You can reassign one of the 8 WIN or 8 DOS devices to these path names and enter new ones and so on.

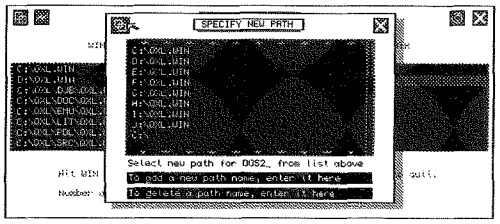
As I have a number of QL CD-ROMs which I keep stored on my hard disk, this program has come in extremely handy for switching between the dozen or so QXL.WINs I have on my hard disk!

QXL.WIN Manager will be available from my website and from most sources of QL free software.

http://homepages.tesco.net/dilwyn.jones/index.html



Main menu display



The path name selection menu

SMSQE v3.07

Wolfgang Lenerz (SMSQ/E Registrar) wrote on the QL-Users mailing list during September:

"SMSQE 3.07 is now ready and being sent to the resellers."

As usual, the sources are available to those who are interested in looking at them and possibly contributing to SMSQ/E development, on Wolfgang Lenerz's website at:

http://www.scp-paulet-lenerz.com/smsqe/

Wolfgang describes the changes:

"There is a new keyword:

EX_M: This behaves a bit like EX in that the calling job continues executing (like EX and contray to EW) but the job created is owned by

the calling job (like EW). This means that if you get rid of the calling job, you will also get rid of the created job(s).
A treat for QXL owners: Thanks to Bruno COATIVY, you can now configure what dos drive letters (C:, D: etc) corresponds to win1_, win2_ etc... Just configure the new SMSQE.EXE file, under the "Host" section.
The Q40 flp_density workaround in v. 3.06 now works. This is a 1 byte change in 1 source file, and I still managed to get it wrong!"

EASYPTR

STOP PRESS - Marcel Kilgus (author of QPC2) is in the process of making contact with Albin Hessler to discuss the possibilities of updating Easyptr for GD2 colours. He has already made some changes largely for his own use and "just for the challenge" he claims, but is not sure if his altered version is releasable as an update. No further information at the time of writing - more news in the next issue I hope!

BASIC Linker

Wolfgang Lenerz

A new version of the Basic Linker exists: V1.25. Mainly, this allows you to use the new colour schemes and system palettes of later versions of SMSQ/E. There were also some bugfixes. You can also set a new configuration item: linker file directory which will be the input directory for linker files only.
As a reminder, the Basic Linker is a program to make writing large Basic files easier by allowing you to split them up in smaller chunks, then linking them, syntax checking them and optionally compiling them.

QDT Progress Report

Jim Hunkins

The good news is that QDT is progressing steadily. The bad news is that it still isn t ready to sell. But if you are fortunate enough to be/have been at the QL2004, you will have had an opportunity to see the latest rev with the updated color management included. And if plans go well, you might even have your very own demo copy of this early version of QDT to play around and get the feel of it with [you will need to already own a copy of menu_rext, the QMenu extensions to use the demo].
The planned demo copy of QDT is time and object limited. You will only be able to open a certain number of folders which each will be limited to a certain number of objects. Also, many of the menu options and most of the Configuration Notebook selections are not yet enabled, primarily since they just haven t been finished yet.
This demo version is definitely intended only to demonstrate the feel of QDT and let you play around with it just a bit. It is NOT intended to actually do much real work with (IE: the idea is to buy the program).
The latest work on QDT ended up being more involved than I was expecting. The big time hits were the update to the color management methods to support the upcoming color theme capabilities and the SMSQ/E color palettes, along with an effort to lock windows down while sub-windows were called on top of them.
The main part of the color work is done but will likely see a lot of bug fixing/tuning when the theme capabilities are rolled in. Also, the window locking out (to avoid a menu from being accidentally covered for example) is primarily there but is a hand done work around since SMSQ/E does not currently have this built in. Some tuning will still need to be done with it.
At this point, most of the primary object functionality is turned on. The next steps will be to finish implementing the folder and icon menu items (mostly just linking to functions already written), and then turning on all the configuration notebook selections, which include everything from screen saver control to desktop resizing to background changes (the fun stuff that you won t normally use everyday but sure makes your desktop easier and more enjoyable).
The plans are to deliver two major items this year. I expect to have an updated demo available with the next version of QL Today. And then, just in time for the holidays, QDT will be available for sale (finally!). Yes, it is far enough along to be able to say this with a high degree of confidence.
Throughout the next year new pieces and enhancements are planned which will be free for the normal upgrade process (web download, bring your original floppy to your dealer, etc). These will include the Theme Manager, an Icon Browser, and a new Jobs frontend.
Please stay tuned as the time draws nearer. I will announce updates on my website:
http://www.jdh-stech.com/ql.htm.

QPAC1 in Hi-Colour

Marcel Kilgus has updated the QPAC1 Utilities now as well, after Tony has dug out the sources. Updates should be available from the known sources J-M-S and QBranch.

Freddy Vaccha

There has been some interest expressed recently via the ql-users mailing list in renewing efforts to have the Perfection word processor updated and re-released. In order to do this, we need to try to contact Freddy Vachha of Digital Precision to see if we'd be allowed to release an updated Perfection to be released via PD libraries.

Does anyone have up to date contact details for Freddy Vachha, formerly of Digital Precision Ltd? The company no longer exists (in fact the name is now owned by Freddy's one time rival QL trader David Batty of Sector Software), and my last contact with Freddy was a brief chance meeting at a tourist attraction some three years ago.

David Gilham has done some work on updating Perfection for high resolution screens and so on, but the copyright situation (especially with regard to the source code) is not too clear, so any help we can get to try to follow this up would be greatly appreciated.

Gee Graphics! (on the QL?) - part 40

H. L. Schaaf

Lame Length?

I've been dabbling into all sorts of interesting things while trying to figure the lengths of Lame curves. I went back to elementary calculus and looked into the "line integral". It sounded like it might be a good way and distracted me into "numerical differention". That turns out to be even more distracting and I'm still playing around with it. I picked up a neat way to have the QL calculate Bessel functions using GRULE, etc., but not sure what to do with them. Can anyone show and tell us when why and how to use Bessel functions?

For Lame curve lengths I'm stuck at the moment with a brute force approach which seems simpler and I think the answers are good to at least 4 digits. The listing 'finitedx_Fun_bas' can be merged with the listing from GG#40 and seems to get us about the right answers for area and length. I'll keep looking for other ways. Do any of you 'numerical analysts' have some ideas?

```
352 REMark finitedx_fun_bas
353 REMark for GG#40 HL Schaaf Sep 22, 2004
355 CLS#0: PRINT #0\\,,'Please wait for estimate of length '
357 PRINT 'Length of curve ? ';finitedx,'Area ? ';sum_area
358 CLS #0
1870 :
1875 REMark to be merged with listing from GG#39
1880 DEFine FuNction finitedx
1890 LOCal j
1900 REMark dydx across x
1910 gap = (b-a)/(210)
1920 gap2 = gap*gap
1930 sum_len = 0
1940 sum_area = 0
1950 FOR j = a TO (b-(gap/4)) STEP gap
1960  sum_len = sum_len + SQRT(gap2 + (f(j)-f(j+gap))2)
1970  sum_area = sum_area + gap * f(j+gap/2)
1980 END FOR j
1990 RETurn sum_len
2000 RETurn sum_area
2010 END DEFine : REMark finitedx
```

I'm reading an interesting book on "Experimental Mathematics" by Jonathan Borwein and David Bailey and wonder if some of the things they are discussing could be cut down to size for use in the QL. Maybe next time?

Crisis 2004

Geoff Wicks

Recently I took a decision that resulted in the deaths of four people. All four were shot dead by police, although this incident did not get into the papers. At the time I was a member of a team taking part in a simulated crisis management exercise. Faced with a situation that was rapidly escalating out of control, we were asked to authorise the first use of live ammunition to control a riot on the British mainland.

Perhaps I should be more honest. I was not actually a member of the crisis control team, but was just pretending to be by participating in an interactive television programme. My career as a powerful government minister ended in ignominy when I pressed the wrong button on the remote control and crashed the system.

With a bump I was brought down to earth and it was time to think again about QL2004. At this event we are looking back on 20 QL years. Technology has changed a lot in those 20 years. When the QL was launched I had just bought my first colour television, and had what was then the luxury of 12 channels. Now we are experiencing an integration of computer, communications and media technology, and if I wanted I could have access to hundreds of channels, many of them interactive. 20 years ago I would be typing this on a portable typewriter, and would send it to the publisher by post. Now I type it in a word processor and send it by email. If my memory serves me correctly, 20 years ago videos and CDs were still expensive new toys and fax, cash machines and mobile phones were yet to make their debut.

The QL has survived for 20 years. Not a bad achievement for a computer most experts wrote off as a failure from day one. In those 20 years it has faced crisis after crisis and constant threats of imminent death. The fact that it is still alive today must say something about the durability and quality of the product. By comparison many dot com companies and products, for all their glossy state of the art technology, have not survived for 20 months.

In the first couple of years after the QL's birth the peripherals industry quickly turned it into a more professional computer, but then major development stopped for about 6 or 7 years. The hardware experts explained that there were good technical reasons why the QL could never address more than 1Mb of memory. There was no point in improving screen resolutions or the number of colours if you had only 1Mb to play with. At its birth the QL had graphics capabilities that were superior to most other computers, but it soon fell seriously behind.

It was not until Miracle Systems challenged the 1Mb assumption, and proved it wrong with the release of the Gold Card that new possibilities began to emerge. Not long afterwards it became possible to emulate the QL on other computers via hardware. Yet more years further on and software emulation became possible. It had taken a long time, but eventually we had faster processors and high resolution screens.

More colours were a long time in coming, but the Q40 coupled with, I believe, financing from the French user group gave the impetus for GD2 colour drivers. Now the new colours are available on all QL platforms.

We have had the GD2 colours for about 5 years, but it is only in the last year or so that they are appearing in software. I suspect that one reason it took so long is that each of the three main QL platforms handles colour in a slightly different way. This has resulted in a mass of technical information that has been difficult for the average QL user (and I suspect some of the experts) to absorb.

Marcel Kilgus has continued to develop the colours and it is now easier to use them in our software. The Window Manager or System colours provide a simple way to use colours that is common to all platforms. Anyone who can write a SuperBasic program, including people who have no interest in the pointer environment, can now program in the new colours. All we have to do is to remember a few simple, new keywords, INK becomes WM_INK etc., and use the right numbers for the colours we want. The Window Manager colours should now become the standard for QL use.

QL2004 is not just looking back, but also looking forward. There are many tasks we still have to do but our numbers have become smaller. How do we face this challenge?

In my opinion one of the most worrying developments in recent years is the increasing passivity of the UK scene. In my article on the new colours in the last edition of QL Today, I referred to the work of many people, but only one was a UK user. As I was writing the article I realised there were over 300 QL users in the UK, but I could count on my fingers the people I knew to be using the new colours actively. Quanta mirrors these passive attitudes. How do we stimulate passive people to become more active?

We also need to think about the traders. Is it any longer realistic to talk of traders when all make a loss from their QL activities? In effect, do we now have only virtual traders? Do we need traders? Can we envisage a traderless QL world?

Yet another worry is the growing gap between SMSQ-E users and Unix/Linux enthusiasts. In the past the QL community has benefited greatly from the skills of the latter. For example, where would we be without the zip and unzip routines Jonathon Hudson ported from other systems? Without these vital tools we would not be able to download programs from QL websites or send programs to one another by email. Yes, emailing and the internet! The Holy Grail that still eludes QL implementation. What future is there for our computer in a society in which media and communications technology are rapidly merging? What are the consequences if we fail to find the Holy Grail?

Earlier this year Peter Graf suggested to me that new QL software such as internet and email access, word processors and image processing could be within our grasp, provided we port software routines from other computer systems. This is one reason we still need Unix and Linux experts within the QL community.

Peter has strong feelings about SMSQ-E licensing which, in his opinion, restricts this development. His opinions have caused much controversy within the QL community, and I have no intention of restarting this debate. To do so would be unproductive and damaging to the future of the QL. (And frankly, I, for one, do not understand all the legal and technical arguments.) However Peter echoes the feelings of many Linux and Unix specialists. Is there any way that we could somehow renew the dialogue with these people, while still retaining the integrity of SMSQ-E?

If not, who is going to write the new software we so desperately need?

Does the QL have a life in the new digital age? Or is should we fall back on the simplicity that is our greatest strength? Over the last year there have been reports of the QL being used to monitor industrial and similar processes. The ease of programming in SuperBasic gives the QL an advantage over more sophisticated computers. Could we use this simplicity to discover new markets and new opportunities?

In few days time I shall abandon the old fashioned QL, reenter the digital society and attempt to relaunch my career as a powerful government minister. I would probably succeed if only that damn remote control was as easy to use as the QL. I fear that once again I shall be brought back down to earth with a bump. Maybe I should just stick to the challenge of QL2004.

GWAIHIR- A QL afloat

Tony Firshman

When I heard Jens Wildgruber was picking Marcel Kilgus up in Brighton from his ocean going yacht, and that he had a Minerva/superHermes equipped Aurora based QL on-board, I just had to have a look.

Roy is a diver, and had cancelled his dive that day as it was (reportedly) too rough. He spent the whole journey to the Marina bemoaning the fact that it was dead calm! We passed millionaire's row - Paul McCartney and others have houses there, and into another millionaire's row - the Marina.

We headed straight for the Weatherspoons pub. Well we had to get our sea legs didn't we!

The Marina was pretty full, and there were some really exotic expensive craft there. We made our way down the slope to the security gate. "All one needs to do is wait a while for someone to go through" said Roy, but Jens was waiting for us.

We passed the motorised dingy that Roy uses for diving (He is a member of the local sub-aqua club) and headed for the boat. Despite all the various exotic sailing and motor craft, Jens' boat had pretty well the tallest mast in the Marina. It is a very large magnificent yacht maybe 40 foot long.

Jens was on his way home after a long round-Britain sail, and was taking Marcel home on the last 2 week leg.

The mechanics of his boat are truly magnificent. He has a pair of what look like brand-new bright yellow diesel engines.

There is a host of electronics, including a very sophisticated self steering gear, with masses of electronic feedback.

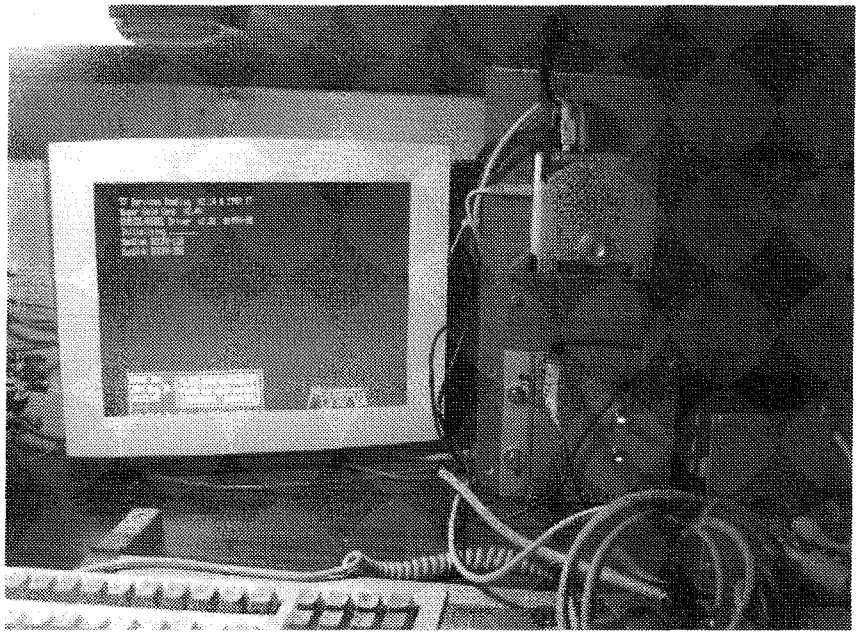
At the heart of his boat is a QL, based on Aurora, Minerva MKII, with analogue I2C circuits, and superHermes. He runs QPC2, and drives an LCD monitor.

His QL does his bank accounts, tasks and duties in the pipeline, books to read etc. More relevant to the boat though, he has a weatherstation (pressure, temperature, humidity) read into the QL by the Minerva I2C analogue interface.

He also reads data from his GPS, "which does nothing at the moment" he says. A Minerva I2C based interface is under construction to read wind force, direction, speed and depth into the QL.

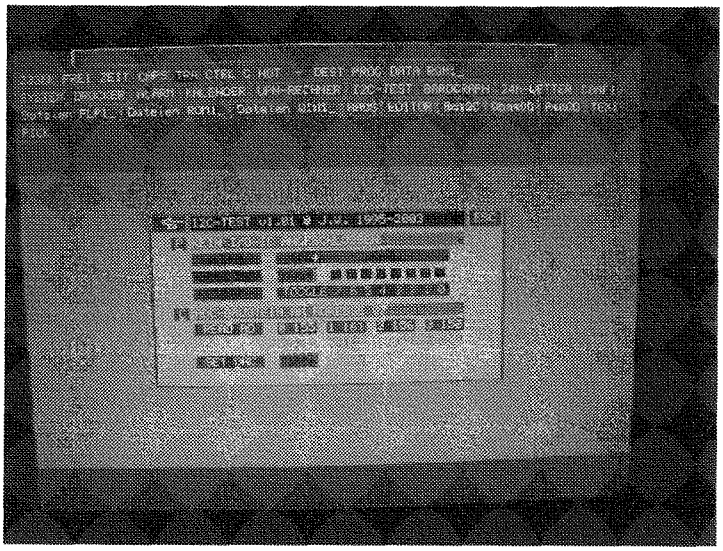
He also has planned:

- Alarm systems (for sudden falling/rising air pressure for instance)
- Battery charge monitoring
- 'Electronic' sea chart. I remember the Dover-Calais seacats have that.

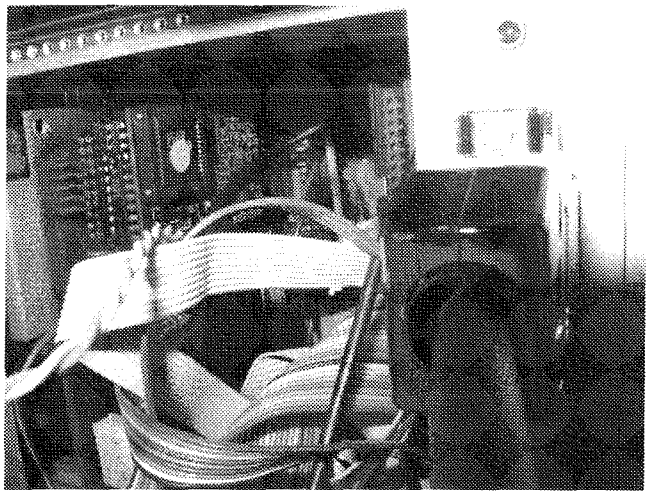


During our inspection he fed us with wine, freshly baked garlic bread, olives and cooked meats.

His email to me finished:
"Thanks for you visiting GWAIHIR and the crew - and meeting the board QL - at Brighton; I'm sorry that I can't talk English - at my school time it was treated like Latin: we translated from English to German and never learned to SPEAK english. So I THINK in German and have to translate then into English, which is a rather slow process and I better can READ English then understand it if it is spoken."



German was not taught at all at my school, so Jens' English is relatively brilliant. I did learn (and fail) Latin though.
Many thanks Jens for your hospitality. It was good to see the I2C programs you gave me, included with my I2C interfaces, in action.



Last Chance to Download

Jochen Merz

As announced in the previous issue, J-M-S is moving. It is quite a long process, and it seems we massively underestimated all the problems. By the time you read this, we've been working for about three months on our new home, hopefully have moved the private bits and carry on working.
The office is planned to be moved by the end of November (yes, quite some time ahead).

It is now clear for several reasons that I cannot maintain the BBS after the move. Reason one is a technical: it is not possible to move more than 10 telephone numbers ... and as we had 14 numbers on two different lines before, some numbers have to go.
While checking and deciding which numbers need to be kept and which ones can go, I went through all the telephone numbers and had a careful look at the BBS situation.

I was planning to move the BBS functionality onto the internet some time ago (you may recall my "Asking for Help" page some issues ago).
The message boxes have completely moved onto email, it is so much easier and cheaper. During the last year there were just a few messages to which I replied quite late ... because I simply forget to log into the BBS.

This is not because of lazyness, it is just that most of the time I check repeatedly, there's nothing and then I tend to forget about it.
Also, time did not allow me to update and maintain it properly (yes, I said this before and it is even more true now).
So, the communication part has moved to E-Mail anyway which seems to be perfect for 99.x% of the users.
Public Downloads - well, not much has happened over the last two years or so (at least). And there are better, other sources anyway, like the download page from Dilwyn.
Infos: same problem. I cannot maintain the pages properly and there is far more up-to-date info available on the web (including my homepage **smsq.j-m-s.com**).

This leaves the initial idea for setting up a BBS: provide updates. As said before, I asked for help to move this facility to the internet as well, and I got a few suggestions (special thanks to Alf Bogner!). The problem was: it was not exactly what I needed or the server did not provide the required features.
What I need is something like the following:
Programming language: ASP
I must be able to create users and a password for every user. And then I "only" need some kind of matrix where I can tick a box which of the files in a certain directory may be downloaded and viewed by which user.
Alf provided a solution via htaccess, but the server does not allow this, unfortunately.
I do not know if this is possible at all, but if anybody has a solution please contact me!

Still, I cannot create public download areas, so here is the last chance to download whatever you may find interesting. Here is a cut-down list in fairly small print, I don't want to fill half the magazine. If you do not have access to the BBS at all anymore, or you would like to have everything in one QXL.WIN file (for QPC or QXL etc.) on CD, then send me an email: **smsq@j-m-s.com** ... if there is interest, I will create a BBS CD which can be obtained for EUR 5,- incl. postage worldwide.

Before I print the list, I would like to write about the history of the BBS - it should give you an idea how much time (and money) went into a system which has been running for over 11 years - permanently! Maintainance (keeping it up-to-date, sorting out problems with new features of new versions of PBOX etc). was really eating up time, but many years ago the QL was my full-time-job.
Unfortunately, things have changed and most of my QL-dedicated time goes into QL Today, as you know.

BBS History

The BBS went online 5th of November 1992. It started on a dedicated ATARI Mega STE, based on QBOX, a BBS software created by Jan Bredenbeek.
A month later the harddisk had to be upgraded to a spectacular 240MB fast SCSI harddisk to improve access time (which it did!) and storage space (this was a lot these days!)
The modem was quickly upgraded to allow up to 16800 connects.
About a year later, a new version of QBOX with improved features was implemented.
Two years later, 1995, the BBS was heavily used and faster modems appeared, so I added a second line with a 33.6 connect and also changed to Phil Borman's new PBOX software ... a completely new system which grew with Phils help to the current system.
He implemented many wishes, multi-language support etc. and I tried to use as many features as possible - "just" the multi-language feature means twice the work!
The PBOX BBS also receives faxes - very useful.
A year later, the computer was changed to an ATARI TT, 10MB RAM and 850MB SCSI harddisk. The serial ports are better and faster, V.34 etc. was now possible.
In 1999 the TT was upgraded to 20MB RAM and a 4GB SCSI drive.
A bit later, the Q40 appeared and I decided it might be a good idea to set the BBS up on a Q40. It was much faster again.
For various reasons I had to move the BBS onto its current and final host machine, a fast PC running QPC2. Here it gives the ultimate speed.
You did not really notice the speed improvement if you dialled in "from outside" between the TT, Q40, QPC2 but when you were sitting in front of the machine, the speed increase was impressive.

Well, if I look at it now: BBSs are quite outdated nowadays, and I think I kept it going just for nostalgic reasons, and because it kept on working and working. But, as nearly nobody uses it, time to close it down.

I really hope to be able to provide an alternative for the update download, but for all the other functions of the BBS, the alternative are better, easier, cheaper and more flexible.

File Requests from JMS Box Duisburg are available 24 hours a day
Files downloaded using a MAGIC filename dont count towards download limits!
Areas from area 30 onwards are not available for all users!

Recognised MAGIC filenames are:

FILES	List of all files on this board
ALLFILES	synonym
FILELIST	synonym

*** Area 3, Mailbox — QDOS: Mailbox Utilities (Archivers ...) ***

QTPI166_zip	18/05/99	434k	[0007] Excellent Terminal Program (XMODEM-ZMODEM...) for the Pointer Environment, V1.66
QFAX285_ZIP	18/05/99	765k	[0002] Latest version of QFAX - complete
UNZ532XQ.EXE	04/04/98	228k	[0011] Latest UNZIP/ZIP (required UNZIP.BAS to extract)
UNZIP.BAS	04/04/98	1k	[0012] Starter for UNZ532XQ.EXE
QPAK_pak	16/09/95	21k	[0004] Automatischer Packer fuer selbstentpackende Dateien (Auspacken mit LRESPR) [HPR]
FaxUtil_zip	17/03/95	3k	[0008] Einige Prozeduren zum Faxe versenden und ansehen..
PCL2G3_ZIP	09/10/94	23k	[0004] Utility to convert Linedesign HP output to G3 (fax) format for pretty faxes
QLReader_ZIP	28/09/94	142k	[0010] Offline message reader for Compuserve, QBox and Internet/Usenet freeware
LFAXV020_ZIP	14/09/94	64k	[0002] Class 1 fax softare for QDOS/SMS.
FSPIC002_ZIP	14/09/94	25k	[0001] Include dithered pictures in QFAX faxes [J. Hudson]
QLTRM232_ZIP	20/12/93	28k	[0002] Neue Version vom QLTERM. Xmodem Bug behoben und 19200 Baud!!!

*** Area 4, Assembler — QDOS: Assembler (Utilities ...) ***

BUTTONFRAME_zip	10/06/96	4k	[0025] How to access the button frame from BASIC
SPRITES_zip	17/02/96	8k	[0016] Alle moeglichen Standard-Sprites. All sorts of standard-sprites (move, resize, F1 .. F10 etc.) [J.Merz]
ICONS_zip	17/02/96	9k	[0011] Die netten Icons aus QD zum Selber-Nutzen. The neat icons from QD. [J.Merz]
AREF_zip	21/10/95	13k	[0007] Assembler Cross Reference (benoetigt I/O2 Toolkit) [HPR]
CPUreport_zip	12/10/95	10k	[0012] Faengt CPU-Exceptions ab und zeigt sie an. Assembler-Source und CONFIG. Traps and display CPU exceptions. With assembler source and CONFIG. [Jo]
ASM20_zip	25/09/95	42k	[0009] 68020 and FPU assembler which generates SROFF Format [A.Rudolf]
GETSTUFFED_ZIP	08/01/95	4k	[0005] Utility to extract the last STUFFER BUFFER contents into BASIC [J.Hudson]
Fpukern_zip	13/09/94	190k	[0004] This is an implementation of the original motorola FPSP Packet.
SYSTEMJOB_zip	07/05/94	5k	[0012] Neue Version (Button Frame). Misst die Rechnergeschwindigkeit und Systembelastung. Mit Assembler-Source. Scheduler speed measurement with Source [Jo]
SCOPY_zip	22/04/94	1k	[0002] Tool to copy screen from $20000 to #ch (e.g. QVME) [E.Ikemann]
UTIL_MENUS_zip	24/12/93	14k	[0000] Different assembler sources which demonstrate how the Menu Extension can easily be used (now even more utils, and some updated). [J.Merz]
DIV_zip	25/08/93	1k	[0003] Division for 32/16 bits [E.Ikemann]
MEXLIB_zip	11/04/93	2k	[0005] Library examiner which uses the Menu Extension [J.Merz]
ButtonUti_zip	21/11/92	3k	[0006] Utilities to allocate, re-allocate and free the Button Frame [J.Merz]
MakeArray_zip	21/11/92	2k	[0006] Utilitiy routine to create a 2-dimensional array for Menus' LIST [J.Merz]
CLSA_E_zip	18/11/92	2k	[0000] Fast screen setup (engl.) [E.Ikemann]
CLSA_zip	16/11/92	2k	[0002] Zur schnellen Screeneinstellung (incl. OUTLINE) [E.Ikemann]
EXLIB_zip	16/11/92	2k	[0007] Filter which tells you which XDEF's are defined in a _REL or _LIB file. [J.Merz]

*** Area 5, SuperBASIC — QDOS: SuperBASIC Programme ***

dbERINNERN_ZIP	04/07/99	5k	[0013] Erinnert an wichtige Termine [Dietrich Buder]
CONFBAS_zip	09/12/94	40k	[0011] flexible approach to create CONFIG blocks for SuperBASIC (works on SMSQ too) [F.Hofsteenge]
Aktdatum_bas	13/10/94	1k	[0004] legt das "amtliche" Datum im Format TT.MM.JJ auf eine Hotkey-Taste.
Arcshell_ZIP	22/04/94	8k	[0005] n'e Shell fuer die Archiver (zoo,zip,arc,lzh) braucht QMENU
TRACE_zip	23/01/94	13k	[0003] Traceersatz fuer Minerva (laeuft auch in Multibasics) sowie neues QLIB_run
DEF2DATA_zip	28/12/93	3k	[0001] Converts EDSPR's files into SuperBASIC DATA Statements [W. Weedon]
BASUTILS_zip	07/06/93	12k	[0005] Diverse BASIC-Utilities mit Source und compiliert [M. Dettwiler]
DicExpand_bas	11/04/93	1k	[0005] Utility which expands a complete QTYP_dictionary into ASCII. [J. Merz]
SPACE_zip	23/01/93	31k	[0005] Extensions to write SuperBASIC pointer based applications quite easily using QPTR [O.Fink]
QPACer_zip	23/01/93	54k	[0005] A marvelous utility for those who wish to set up QPAC2 BOOT files, but don't have the knowledge [J. Davies]
DRACHE_bas	05/12/92	1k	[0001] Draws a Dragon Curve
TK2_Anleitung_zip	05/12/92	133k	[0000] Deutsche SuperToolkit II-Anleitung
Merge_bas	16/08/84	6k	[0006] A program to merge two or more Toolkits [P. Borman]

*** Area 6, ProgLang — QDOS: Programmiersprachen / Programming ***

PERL4QQDOS_ZIP	04/07/99	1270k	[0005] Perl for QDOS
422HLIBC_ZIP	12/04/98	58k	[0003] C68 4.22h libc library
C68BINQ_ZIP	12/04/98	70k	[0005] C68 binaries (4.22 required)
422FSRC4_ZIP	12/04/98	576k	[0003] C68 4.22f source of extra libraries
422FSRC3_ZIP	12/04/98	632k	[0003] C68 4.22f source of main C library
422FSRC2_ZIP	12/04/98	350k	[0003] C68 4.22f compiler phases and utility programs
422FSRC1_ZIP	12/04/98	546k	[0004] C68 4.22f source of the main compile phase
422FDOC2_ZIP	12/04/98	186k	[0003] C68 4.21a libraries and utilities documentation
422FDOC1_ZIP	12/04/98	242k	[0003] C68 4.21a main compiler programs and general documentation
422FRUN3_ZIP	12/04/98	358k	[0003] C68 4.22f extra utilities and libraries
422FRUN2_ZIP	12/04/98	216k	[0003] C68 4.22f boot files and utilities
422FRUN1_ZIP	12/04/98	268k	[0003] C68 4.22f main system disk with compiler, header files and libraries
C68TOOL_ZIP	23/12/97	38k	[0002] C68 Benutzer-Oberflaeche [T. Roesner]
qmenuC_zip	03/12/97	90k	[0010] Use the Menu Extension from "C"
readme_doc	11/12/95	9k	[0001] A short explanation for all Z88 related files.
OZdefc_zip	11/12/95	18k	[0001] The standard Z88 operating system manifest definition files.
Z80lib_zip	11/12/95	86k	[0001] A standard library of routines for use by Z88 applications. With source.
Z80src_zip	11/12/95	145k	[0002] The complete source for the native Z88 module assembler
devnotes_zip	11/12/95	260k	[0001] The Z88 Developers' Notes V3 (PipeDream format) - how to program Z88...
QLz80asm_zip	11/12/95	176k	[0002] The Z80 (cross) Module Assembler with ANSI C source files and documentation
TTOOLS_zip	14/06/95	229k	[0003] Texttools 1.1. Alles was GNU Text nicht hat und F.Krojer trotzdem fuer nuetzlich haelt.
RECODE_zip	14/06/95	166k	[0002] GNU Recode 3.2.4. Zeichensaetze verschiedener Betriebssysteme und Darstellungsformen konvertieren. Erweitert um QDOS und TOS [F.Krojer]
GNUTEXT_zip	14/06/95	666k	[0002] GNU Text Utilities 1.3 - nicht nur uebersetzt, sondern sie funktionieren auch [F.Krojer]
BRFORTH_zip	13/07/94	102k	[0003] Brouhabouha Forth Interpreter & Compiler
MAKE_CPP_zip	03/09/93	51k	[0000] Make und CPP angepasst fuer die QXL Karte
C68EXAMPLE_zip	07/07/93	8k	[0001] Demo-Programm fuer EASYPTR3, compilierbar mit C68

QDrexx_zip	21/06/93	3k	[0002] REXX interpreter driver Thing for QD [O.Fink]
REXX_zip	28/05/93	128k	[0003] Neue Programmiersprache, inclusive Dokumentation.
QDC68_zip	28/05/93	3k	[0005] C68 Compiler driver Thing for QD5 [O.Fink]
QDpas_zip	28/05/93	3k	[0001] ProPascal Compiler driver Thing for QD5 [O.Fink]
C68DESK_zip	14/05/93	26k	[0003] Benutzeroberflaeche fuer C68 Compiler

****
* Area 7, Utilities QDOS: Generelle Utilities / General Util *
****

ue4upd15_zip	14/09/00	408k	[0001] MicroEMACS v4.00 update to 13/09/00 release: now with a vertical scroll bar. 5thierry Godefroy)
FI2v3e41_zip	13/08/00	175k	[0005] FileInfo II v3.41 (english): bug fix release (Thierry Godefroy).
ctags403a_zip	12/08/00	251k	[0003] Tags file generator for use with MicroEMACS v4.00: now knows about SBASIC sources ! (Port by Thierry Godefroy)
ue4upd14_zip	12/08/00	406k	[0002] MicroEMACS v4.00, update to 08/08/00 release: corrects a bug in tag handling. (Thierry Godefroy)
ACP4e01_zip	29/04/00	67k	[0006] ACP v4.01 (english version): a small bug and a typo fixed. (Thierry Godefroy)
ZOO_zip	28/01/00	56k	[0006] Zoo compress/uncompress for ACP
ue4upd13_zip	05/01/00	406k	[0004] MicroEMACS v4.00 update to 03/01/00 release: now 100% Q40-compatible. (Thierry Godefroy)
bzip2-095d_zip	01/01/00	648k	[0002] bzip2 v0.9.5d (30/12/99 release): 100% ACP v4.00 compatible release. (Port by: Thierry Godefroy)
ctags333_zip	01/01/00	214k	[0000] Exuberant ctags v3.3.3: for use with MicroEMACS v4.00 ("tags" files generator). (Port by: Thierry Godefroy)
gzip124c_zip	01/01/00	317k	[0003] gzip v1.2.4 (30/12/99 release): 100% ACP v4.00 compatible release. (Port by: Thierry Godefroy)
ACP4e00_zip	01/01/00	67k	[0002] ACP v4.00 (english release): now with support for tar/gzip/bzip2/compress and a new "tools" menu allowing to split big archive files (or any file) into several fragments and glue them back later. (Thierry Godefroy)
tar005_zip	01/01/00	111k	[0001] tar v0.05: 100% ACP v4.00 compatible release. (Port by Jonathan Hudson, Fixes and improvements by Thierry Godefroy)
FI2v3e40_zip	13/10/99	165k	[0004] FileInfo II v3.40 (full english release). (Thierry Godefroy).
emacs400_zip	03/01/99	722k	[0001] MicroEMACS v4.00 (02/01/99 release): with PE menus and support for FileInfo II, QMenu (file selector and scrap) and Adrian Ives' "The Shell". QDOS/SMS port by: Thierry Godefroy.
crib_zip	06/11/98	190k	[0006] Geoff Wicks' public domain spelling crib program.
sh108_zip	04/04/98	86k	[0003] UNIX-style Shell v1.08 (New DO command)
ToHTML101_zip	04/04/98	15k	[0006] Filter: Convert plain text to HTML
unfold102_zip	24/03/98	17k	[0007] Filter to remove hard line breaks from text files
ls102_zip	21/03/98	20k	[0005] External LS command for Shell v1.02 (Update/bug fix)
Basc112a_zip	17/03/98	42k	[0001] Latest version of Basconfig - the program which puts CONFIG blocks into QLiberated programs
Got100_zip	08/03/98	19k	[0006] What GOT passed to a Job?
ShLines030_zip	08/03/98	61k	[0004] Show what lines in a file contain a string
TidyAsm030_zip	08/03/98	59k	[0003] Reformat _asm and _dis files
Set100_zip	08/03/98	16k	[0007] External "Set" command for the Shell (environment variables)
Drive_zip	08/03/98	22k	[0004] Command Line Utility for linking/unlinking QUBIDE Partitions
Encode080_zip	08/03/98	58k	[0005] Simple File Encoder/Decoder
QLToolbox98_zip	08/03/98	449k	[0007] Latest (Shell Compliant) versions of all my utilities in one handy modem-stressing archive (Adrian Ives)
TypeInFile025_zip	23/02/98	62k	[0004] Type the text of a file into other applications (needs MENU_rext)
FComp025_zip	23/02/98	54k	[0003] Simple file comparator (needs MENU_rext)
Find045_zip	23/02/98	58k	[0003] Find strings in file (needs MENU_rext)
ListNames045_zip	23/02/98	58k	[0003] ListNames v0.45
WhatKey045_zip	23/02/98	43k	[0004] What is the ASCII code of THAT key?
CompareDirs030_zip	23/02/98	58k	[0004] Compare contents of two dirs/devices
DirTools030_zip	23/02/98	105k	[0004] Update: DirList v0.30 MakeDirs v0.75
EditFName_zip	09/02/98	16k	[0004] Interactive Filename Editor (Needs Menu_Rext)
UNPIC006_ZIP	01/11/97	227k	[0019] Tools to convert _scr and _pic graphics to PNG, PS, GIF, TIFF, BMP or PCX formats.
QPAC2patch_zip	16/08/97	11k	[0009] QPAC 2 patcher: makes QPAC 2 fully FileInfo II v3.0 compatible. (Thierry Godefroy)
scrview_zip	27/05/97	30k	[0010] SCR Viewer for Pointer Environment [A. Carpi]
DIYTK_ZIP	07/04/97	1188k	[0014] The complete QL World DIY-Toolkit
hdql_zip	29/10/96	12k	[0010] macht aus formatierter DOS HD Disk (wie gekauft) eine QL HD Disk
QPCthing_zip	05/10/96	6k	[0012] QPC extensions thing v1.00: Adds VGA color palette support to QPC v1.01+ (Thierry Godefroy).
NetERROR	23/09/96	4k	[0009] Netzwerkfehler beheben
HTMLmach_zip	15/09/96	24k	[0017] Small utility program to write html docs. Requires Menu_rext and Qlib_run [Roy Wood]
Birthday_zip	13/09/96	62k	[0013] Remember important ays (Deutsch & English) [R. Fieseler]
QFormat_zip	10/08/96	18k	[0015] Quick re-format for disks [N.Dunbar]
Utilities_zip	20/04/96	17k	[0019] Various utilities, incl. an ANSI code stripper [P.E.Forssen]
BAUDRATE_zip	12/04/96	56k	[0018] Baudrateneinsteller (mit Config einzustellen)
DATAprinter_zip	21/03/96	35k	[0012] Small utility for printing envelopes from a Datadesign file(provided).Needs menu_rext, Datadesign engine and pfDATA. By Roy Wood
Viewer_zip	21/03/96	210k	[0014] Dilwyn Jones' text viewer and screen grab facility version 1.10. Includes function to use screen dumps in the text.
TELEFON203b_zip	21/01/96	75k	[0011] Pointergesteuertes Telefontimer Programm, fuer alle ohne Gebuehrenimpuls.
stfx010_zip	17/01/96	8k	[0010] Faxen mit QD
CSC_zip	07/01/96	2k	[0007] Filter to convert textfiles from qdos to tos/dos and vv. [E.Ikemann]
trapper_zip	04/01/96	6k	[0003] EXeptions Report von J. Hudson faengt Programme ab die sich schlecht benehmen.
TEST909_ZIP	21/10/95	31k	[0006] Speed-Test mit einigen Verbesserungen und neuen Daten!
RecoverQ_ZIP	16/10/95	103k	[0008] 100% Disk-Kopierer (benoetigt I/O2 Toolkit) [HPR]
NorBack404_zip	23/07/95	89k	[0006] Backup program (Needs Menu_rext) Bugfix (A.Borretzen & P.Monstad)
ungif94_zip	09/04/95	107k	[0017] Ungif 0.94
engif24_zip	09/04/95	5k	[0009] Engif 0.24
CSM116_ZIP	09/04/95	32k	[0003] CSM 1.16 for QTPI 1.52
DOS_zip	25/02/95	2k	[0004] A DOS like font (eg for QD) [E. Ikemann]
jpeg_zip	16/02/95	89k	[0013] *.JPG to *.GIF changer [from QITALY by E.Ikemann]
env_bin	03/02/95	1k	[0011] modified env_bin for QXL Version 2.47 (thanks to D.J.Walker)
clavier_zip	24/01/95	84k	[0007] Software for the QXL: allows you to reconfigure every key on the keyboard
imgtopws_zip	10/01/95	82k	[0003] IMG to PE format changer by Jon Slater [E. Ikemann]
Cal_zip	25/11/94	141k	[0001] Datums-Funktionen etc. (mit C-Source) [F.Krojer]
Fortune_Zip	25/11/94	35k	[0003] Spruechesammlung zur Zufallsauswahl (mit C-Source) [F.Krojer]
ESCP2HC102_zip	27/10/94	9k	[0006] New Version of ESC/P2 Hardcopy Routine (left margin can be set) [E. Ikemann]
GHOSTSCRIPT_zip	14/09/94	584k	[0007] Postscript Interpreter for QDOS.
NENS13_zip	12/07/94	73k	[0003] Neuer, sehr kurzer Text-to-Postscript Konverter mit C-Sources.
GIFTIFF_zip	15/06/94	57k	[0015] Konvertiert QL-Bilder in TIFF und GIF Format [J.Hudson]

SUB_zip	05/06/94	4k	[0004] SUBstitute Device Driver [Phil Borman]
WatchMem_zip	31/03/94	30k	[0006] Nuetzlicher System-Ueberwacher fuer GoldCard und ATARI's [Ian Bruntlett]
NEWS_DIARY_zip	24/12/93	2k	[0000] A Diary program for QPac1's Calendar. Requires QD. [O.Fink]
QSIDeutsch_zip	13/11/93	5k	[0001] QSI in Deutsch und als PRG-File zum Anschauen und Aendern.
RESQL_zip	05/11/93	31k	[0007] Useful and effective Disk/File repair utility.
QSI_zip	05/11/93	11k	[0005] Speed Index program [Eros Forenzi]
PSIONMOD_zip	05/11/93	13k	[0008] Powerful patches for the Psion programs
FASTPOKE_zip	29/10/93	6k	[0002] Button to switch GoldCard to 24MHz
WINMON_205_zip	13/02/93	6k	[0000] Harddisk monitor (German & English) [E. Ikemann]
FREEMEMBT_zip	23/01/93	6k	[0010] A neat little button that shows the rest of your free memory [O. Fink]
PICKER_zip	23/01/93	4k	[0001] A small job that picks all jobs (quite useful if set on a hotkey) [O. Fink]
FILESPR_zip	23/01/93	29k	[0001] A utility to convert QPTR sprite definitions (basic or assembler) into the easySPRITE format [O. Fink]
HD_UTILS_zip	17/11/92	3k	[0000] Some utilities which help in maintaining backup copies from harddisk to harddisk or changeable etc. [Jochen Merz]
FILTER24_zip	15/11/92	2k	[0002] Get proper 9pin-printer output on 24pin printers [Jochen Merz]

****

* Area 8, Pointer QDOS: Pointer Environment Utilities & Pr *

****

EasyCircuit_zip	02/02/00	174k	[0002] EasyCircuit [A. van Dällen]
GV115_ZIP	18/05/99	100k	[0026] Graphics Viewer and SDUMP Front End V1.15 [Dilwyn Jones]
ACP3e5ok_zip	09/01/99	56k	[0012] Well, it seems i made AGAIN a mistake when building the ACP35fix_zip archive (the OLD bad linked binary was archived instead of the new one). So here is the full ACP 3.5e package, with the GOOD binary in it (PROMISSED, this time I checked three times !)... I would kick my ass sometimes. (Thierry Godefroy)
PEIGuide_ZIP	26/09/98	86k	[0004] Pointer Environment Idiots Guide
QASCADE_ZIP	13/04/98	54k	[0013] Latest Qascade which does not require ENV_BIN anymore (J.Hudson)
QEYES_ZIP	12/12/97	27k	[0008] The famous eyes which follow your pointer [J.Hudson/P.Borman]
scratch33_zip	03/12/97	48k	[0023] Extremely useful program to cut areas from screen, save them, stuff text into programs etc. [Phil Jones]
ACP3e4_zip	16/08/97	54k	[0011] Archivers Control Panel v3.4e (full english version): corrects a bug that could make QDOS/ARGOS system to crash under some circumstances. (Thierry Godefroy).
pex32_zip	24/07/97	28k	[0012] PEX V1.32
FILESELECTOR_ZIP	24/07/97	20k	[0012] A Different way to use the QPAC 2 files menus. Needs QPAC 2, Menu_rext and QLib_run to work. V1.04 [Roy Wood]
VATcalc_zip	10/06/97	16k	[0001] Small calculator for V.A.T. sums. by Roy wood
Histiv27_zip	22/09/96	24k	[0007] Hierbei handelt es sich um eine nochmals etwas berarbeitete Version der HISTORY [Boris Jakubith]
PROCMAN_zip	10/08/96	20k	[0016] Procedure and Function Manager, requires MENU [D.Jones]
PEK_zip	04/08/96	303k	[0018] SBASIC Pointer Environment Kit [John Miller]
CACHE_ZIP	20/09/95	1k	[0006] Is a button witch indecates the cache (on or off) [E. Ikemann]
CUT-VIEW_zip	10/06/95	3k	[0006] A tool to have a view of PE pictures. [E. Ikemann]
DocView100_ZIP	22/11/94	34k	[0009] Utility to view Quill Doc files [P. Monstad]
KEdit_ZIP	06/10/94	25k	[0003] Keyboard table editor for ATARI Emulator & SMSQ [Pal Monstad]
MultiDos_ZIP	16/07/94	33k	[0006] QDOS & SMS auf DOS Dateien Konverter [T. Godefroy]
Librun335mod_zip	29/05/94	10k	[0004] QLib_run v3.35mod: debugged version needed by all progs from T. Godefroy.
Search221_ZIP	23/05/94	27k	[0007] If you cannot remember the complete filename of a file anymore ... [A. Boerretzen]
SFormat1E5_ZIP	23/05/94	23k	[0003] Super Formatter [T.Godefroy]
CALC1E3_ZIP	23/05/94	24k	[0003] Toller Taschenrechner [T.Godefroy]
LaunchPad_ZIP	22/04/94	7k	[0002] QPaccy style Launch Pad. [O.Fink]
HandyMenu_ZIP	05/11/93	26k	[0012] Useful additions to QPAC 2 [Ian Bruntlett]
DemonsCave_ZIP	10/08/93	5k	[0003] Release the demons now into your QDOS system. [O.Fink]
Fonted_ZIP	23/01/93	37k	[0007] Sophisticated font editor [O. Fink]
Colours_ZIP	23/01/93	4k	[0000] Gives the code number of colour/stipple combinations [O. Fink]
Convert_ZIP	23/01/93	5k	[0002] Convert numbers into decimal/binary/hex or ascii form [O. Fink]
SysInfo_ZIP	23/01/93	6k	[0007] Reveals the current state of your system [O. Fink]
ScanTree_ZIP	23/01/93	13k	[0007] Read and re-create Level 2 directory trees (very useful for harddisk backups) [O. Fink]

****

* Area 9, Various QDOS: Sonstige Software / Various Softwa *

****

POV31gRun_zip	24/08/99	1301k	[0002] Persistence Of Vision ray-tracer v3.1g with support for Aurora 256 colours and Q40 65536 colours modes. (Ported by Thierry Godefroy).
POV31gSrc_zip	24/08/99	581k	[0001] POV v3.1g sources.
XCHDOCS_ZIP	17/05/98	221k	[0005] Xchange manuals
XCHANGE_zip	17/05/98	269k	[0006] XCHANGE 3.90L (mit allem Drum und Dran)
squareescher_zip	03/12/97	215k	[0008] Escher Knot [Herb Schaaf]
spnsphspi_zip	03/12/97	205k	[0010] Spinning Sphere [Herb Schaaf]
ribnescher_zip	03/12/97	115k	[0009] Escher Ribbon [Herb Schaaf]
rdsrol_zip	03/12/97	106k	[0006] Random Dot Stereogram [Herb Schaaf]
rdsc_zip	03/12/97	36k	[0005] Random Dot Stereogram [Herb Schaaf]
rds_zip	03/12/97	9k	[0005] Random Dot Stereogram [Herb Schaaf]
eschpin_zip	03/12/97	202k	[0006] More Escher graphics [Herb Schaaf]
T87PATCH_ZIP	02/06/97	10k	[0004] Text87 Patch for Aurora
CBACK17b_zip	27/04/97	29k	[0010] Disk/File Backup, BASIC-Programm und Hintergrund-Job
qltoday6_zip	17/03/97	49k	[0014] Working ZIPs for QLTODAY Cover Disk Issue 6 (probably damaged on a number of disks)
Uhr_zip	08/09/96	12k	[0022] (m)eine kleine bahnhofsuhr.
MANDEL143_ZIP	31/08/96	297k	[0002] Mandelbrot Generator [Dietrich Buder]
QLWorld_zip	12/06/96	108k	[0005] QL World Index in DataDesign format
DIGI_zip	21/10/95	136k	[0006] Simulationsprogramm fuer digitale Schaltungen (benoetigt I/O2 Toolkit) [HPR]
PAINT_zip	01/07/94	18k	[0006] Slightly improved PAINT program (from QPTR, which is required) [O. Fink]
MicroEmacs_zip	15/06/94	148k	[0002] Der beruehmt-beruechtigte Editor [R.Kettlewell]
SPECT170_zip	14/03/94	117k	[0000] Spectrum 48/128 Emulator Spectator V1.70
BIO_zip	19/01/94	34k	[0002] Biorhythmus unterm Pointer Environment [F.Schwanitz]
ROSES_zip	05/12/92	3k	[0007] ... und noch ein schoenes Demo!
EYES_zip	05/12/92	2k	[0006] Schnuckeliges Demo [John Dent]
HYPER_zip	11/09/84	3k	[0007] Korrigiertes HYPER von m. Wiemers (geht jetzt auch mit SMSQ) [E. Ikemann]

****

* Area 10, Games QDOS: Spiele / Games *

****

PuzzleDreieck_zip	05/08/98	19k	[0010] Ein kleines, kniffliges, dreieckiges Puzzle!!! Neue Version! [F.D.]
SAME_ZIP	20/07/98	16k	[0004] Neues tolles Spiel/New great game [F.G.Dibwoski]
Mines_zip	13/02/98	35k	[0009] Findet alle Minen auf einem Feld! (Benoetigt PE) [F.D.]
CrazyCard_zip	25/08/96	17k	[0008] Pointergesteuertes Kartenspiel [F.Dibowski]
Pegged_zip	25/08/96	14k	[0006] Pointer-gesteuerter Logig-Trainer [F.D.]
Reversi_zip	20/08/96	22k	[0010] Das altbekannte Reversi umgesetzt auf PE! [F.D.]
Bullets_zip	07/08/96	20k	[0007] Ein kleines, kniffliges NIMM - Problem!!!!! [F.D.]
goban_zip	07/06/96	28k	[0015] Pointer driver version of GO
DECIPHER_zip	21/03/96	22k	[0010] Pointergesteuerter Logiktrainer [F.Dibowski]
ExplAtoms_zip	23/12/95	20k	[0009] Elektronen muessen so gesetzt werden,dass der Gegner vom Spielfeld fliegt. PE

RubiksAREA_zip	20/10/95	19k	[0007] PE-Programm benoetigt Gehirnschmalzzzz!
Games_zip	05/11/94	31k	[0007] Games von w.Lenerz (qltris,memo,puzzle,lucas) Pointergesteurt !
Advent_zip	06/06/94	79k	[0005] Adventure-Spiel (englisch) mit C-Sources. [Erling Jacobsen]
K&Q_zip	15/04/94	20k	[0005] Spiel
Quzzle_zip	04/08/93	11k	[0009] Spiel fuer's Pointer Environment (man achte auf 3D-Button) [R. Weeks]

****

* Area 11, text87 QDOS: Text87 *

****

scrfonts_zip	07/04/97	7k	[0015] Proportionale Helvetica Fonts fuer Text87
TTC_zip	03/07/95	5k	[0004] Ein Prog. zur Uebersicht von P87-Treibern [E. Ikemann]
Helvetica_zip	03/07/95	7k	[0009] Font
Small_zip	03/07/95	2k	[0011] Smallest possible (visible) font ideal for condensed fonts and super-/ subscript [Jochen Merz]

****

* Area 12, Demo QDOS: Demo Versionen / Versions *

****

QPC2DEMO_zip	25/02/00	215k	[0010] QPC 2 Demo V1.54 (does not write to FLP/WIN)
GWDEMO_zip	17/12/97	221k	[0016] Latest versions of Geoff Wicks demo programs, solvit, Thesaurus and Style Check. Including the new pointer driven Thesaurus.
QSpreadDemo_zip	23/05/97	87k	[0027] QSpread V1.41 Demo - jedoch ohne Speichern und Drucken! Demo - no print and save!
ZEXCEL_zip	16/03/95	177k	[0002] ZX Spectrum Emulator (Demo Version) [Ergon]
QLM_zip	14/04/94	60k	[0002] Q-Library Manager
FDU_zip	14/04/94	71k	[0004] Floppy Disk Utilities
DEA_zip	14/04/94	96k	[0005] DEAssembler
OWR_zip	14/04/94	95k	[0003] Open World
CueshellD_zip	07/03/94	28k	[0006] Deutsche demo Version von CueShell
CueshellE_zip	07/03/94	28k	[0005] English demo version of CueShell
LDesDEMO_zip	22/02/94	310k	[0006] LineDesign Version 2 (alles geht, beim Ausdruck wird aber immer ein Schatten-Logo mitgedruckt).

****

* Area 13, LineDesign QDOS: LineDesign fonts, clipart etc. *

****

Clip9_zip	14/04/96	122k	[0015] Clipart for LineDesign #9
Clip8_zip	14/04/96	308k	[0013] Clipart for LineDesign #8
Clip7_zip	14/04/96	284k	[0010] Clipart for LineDesign #7
Clip6_zip	14/04/96	285k	[0008] Clipart for LineDesign #6
Clip5_zip	14/04/96	287k	[0011] Clipart for LineDesign #5
Clip4_zip	14/04/96	294k	[0008] Clipart for LineDesign #4
Clip3_zip	14/04/96	254k	[0008] Clipart for LineDesign #3
CLip2_zip	14/04/96	475k	[0008] Clipart for LineDesign #2
Clip1_zip	14/04/96	577k	[0009] Clipart for LineDesign #1
Fonts8_zip	14/04/96	359k	[0008] Fonts for LineDesign (and ProWesS) #8
Fonts7_zip	14/04/96	407k	[0002] Fonts for LineDesign (and ProWesS) #7
Fonts6_zip	14/04/96	407k	[0002] Fonts for LineDesign (and ProWesS) #6
Fonts5_zip	14/04/96	383k	[0002] Fonts for LineDesign (and ProWesS) #5
Fonts4_zip	14/04/96	402k	[0004] Fonts for LineDesign (and ProWesS) #4
Fonts3_zip	14/04/96	373k	[0005] Fonts for LineDesign (and ProWesS) #3
Fonts2_zip	14/04/96	421k	[0010] Fonts for LineDesign (and ProWesS) #2
Fonts1_zip	14/04/96	402k	[0011] Many fonts for LineDesign (and ProWesS) #1
README_TXT	14/04/96	10k	[0018] Explanation to fonts and clipart

****

* Area 14, SBASIC SMSQ: SBASIC Programme / SBASIC programs *

****

DicoLis_bas	25/05/97	3k	[0002] update of dicexpand (more general version ?) with a solution to wait langage b
QREF_PATCH_bas	01/09/96	1k	[0000] Patch for QREF to display 32 bits in the address field [E. Ikemann]
SUB_bas	02/04/96	1k	[0023] Oeffnet einen Button fuer die Einstellung des SUB-Devices von [W. Strate]
UMLAUTE_zip	21/03/96	2k	[0015] Wandelt deutsche Umlaute DOS ‹-› SMSQ/E. [J.Merz]
ADDM_bas	31/12/95	2k	[0014] Add own modues to an SMSQ/E file [T. Tebby]
SCAN_bas	28/10/95	1k	[0023] List all the modules in an SMSQ/E file [T. Tebby]
LANG_bas	28/10/95	12k	[0019] This shows you how to create own language-dependent tables [T. Tebby]
PAINT_zip	19/09/94	19k	[0008] Modified version of PAINT which can be executed directly (QPTR required) [J.Merz]
PACK_zip	19/09/94	3k	[0018] Small utility which makes the use of ZIP, ARC and LHQ easier. [J.Merz]
COPYDIR_zip	19/09/94	2k	[0020] Creates the same set of subdirectories on another harddisk. [J.Merz]

****

* Area 24, Text Interessante Texte / Interesting Docs *

****

RTF_zip	19/11/96	60k	[0012] RTF description
qhj25_txt	17/08/96	29k	[0005] QL Hackers Journal #25 by Tim Swenson
qhj24_txt	17/08/96	19k	[0004] QL Hackers Journal #24 by Tim Swenson
ansisys_txt	27/05/96	13k	[0001] Codierung der Steuerbefehle (nach "ANSI.SYS" bekannter Pe-Zeh-Kisten)
ansiterm_zip	10/05/96	7k	[0000] VT100/VT52 und ANSI-Terminal-Steuercodierung (nicht ganz vollstaendig)
kohl_txt	22/03/96	57k	[0005] ... und viele Kohl-Witze - ganz lustig!
MANTA_TXT	22/03/96	52k	[0002] ... und 'ne Menge Manta-Witze - auch neue Ausgabe!
BLONDINEN_txt	22/03/96	119k	[0004] Haufenweise Blondinen-Witze - neue Ausgabe!
qhj23_txt	27/02/96	28k	[0001] QL Hackers Journal #23 by Tim Swenson
qhj22_txt	27/02/96	29k	[0002] QL Hackers Journal #22 by Tim Swenson
ZMODEM_zip	10/02/96	14k	[0003] Documentation of the ZMODEM protocol (english)
QHJ_zip	16/11/95	231k	[0003] QL Hacker Journal 1 - 21
PiQdsk_zip	12/11/95	131k	[0002] Begleitende Dateien zu Progr. in QDOS [HPR]
PiQdoc_zip	12/11/95	256k	[0002] Programmieren in QDOS - alle Textdateien als DOCs [HPR]
C_FAQ_zip	12/07/94	54k	[0000] Frequently asked questions about C
MODEMFAQ_zip	05/06/94	34k	[0006] Frequently asked questions about Modems, Protocols etc. INTERESTING!
Fax_Class2_zip	22/04/94	16k	[0000] Class 2 fax command description
Fax_Class1_zip	22/04/94	8k	[0000] Class 1 fax command description
GOOFS.LZH	22/04/94	39k	[0001] GOOFS, Pleiten und Co aus bekannten Filmen. Aus MAUS Netz. [N. Roller]
ERRORS_zip	17/01/94	21k	[0002] Lustige Fehlermeldungen, z.B. "No Keyboard present, press F1" [N. Roller]
ZMODEM2_zip	03/04/93	36k	[0002] Another comprehensive ZMODEM protocol documentation (english)
ZMODEM_zip	03/04/93	14k	[0001] Another ZMODEM protocol documentation (english)
YMODEM_zip	03/04/93	19k	[0000] YMODEM protocol documentation (english)
XYMODEM_zip	15/11/92	37k	[0002] Documentation of the XMODEM and YMODEM protocols (english)

As said before, this is a cut-down list of the public areas of the BBS. Most items are rather old, some are in German, but if you think you want something from the BBS, download it now or get it on CD.

I would like to thank everybody who helped supporting the mailbox, especially Jan and Phil for having written the BBS software and for their support!

Wares

John Perry

There's heaps and heaps of computer software out there, much of it free, for all sorts of computers old and new. I have got into the habit of downloading programs for both PCs and QLs and find that as far as the internet is concerned at least, the PC and QL have a lot in common in terms of the terminology used. Having got confused with the various kinds of *wares when it comes to software, I made the effort to make sure I knew what they all were and thought I'd share my information with readers.

Software usually falls into these broad categories:

- Commercial software - software you can buy or generally pay for.
- Shareware - try it out free, register and pay if you like it and wish to keep using it
- Freeware - free software which can be freely copied, but the author retains copyright.
- Public Domain - software which the author has made freely available and you can usually do what you like with it (legal, decent, honest and all that of course!). This is usually copyright free and you'd be able to modify it, include code in your own programs and so on, although there may be a few terms and conditions limiting what you can do with it.
- Charityware - software which can usually be freely copied and distributed, but where the author or publisher has requested that a donation be made to charity if you like the software and wish to keep using it.
- Cardware - if you like the software, send the author a postcard (on the QL, Simon Goodwin's DIY Toolkit is an example of this, which reminds me I haven't yet sent him a postcard!).

Some of the above categories may be modified by including specific terms and conditions to define the distribution of the software. For example, you may pay to buy a program and find when you read the small print that what you have bought is a licence to use it rather than purchased ownership of the software as such.

Charityware may be accompanied by text which states that the donation to charity is voluntary, in other words, if you like the software and keep using it, you are asked to consider making a donation to charity, but you don't have to.

Shareware can be a difficult one. Sometimes you are asked to pay if you wish to keep using it. Sometimes you pay for software support rather than the software itself (which may be completely free). Sometimes the author only requests that you make a payment, it is actually up to you, although if someone has graciously made available a great piece of software which you use regularly, I expect many of us would send a little payment to the author in case we ever needed to ask for help or just to reward him/her for their work in an attempt to make sure he/she releases more great programs! Sometimes you are allowed to use the 'free' version but are invited to pay to get an enhanced version of the program. The free program may have an advert such as an opening or closing screen which says something like:

"Thank you for using Fred's Database. This version is free. For the small sum of $10 you can get a much enhanced version of this program with loads of new features! Visit* freds.software@hisplace.com *for more details!"

The moral of all this is to read the text files and small print which come with the program. I have installed software which asked for payment and after using it a couple of times felt guilty that I might be using it illegally by not paying and continuing to use it. Then I find that in fact payment is voluntary in some cases. I've got into the habit of paying for something I find really useful even if payment is voluntary. Sometimes I've been pleasantly surprised to receive an update for the program either by email (or less often) on a disk or CD in the post. On the other hand, I tend not to pay if I only use the program once and remove it from my computer before the guilt takes over! It can be difficult if you have to pay in a foreign currency, especially if you have to send cash or cheques abroad, although the modern internet facilities for paying everything by card electronically over the net makes life a lot easier.

When you download a program from the web, you'll often find the program has been archived and compressed into one file, often using a program called ZIP. There is a version of ZIP for the QL, maintained by Jonathan Hudson, and available from his website at **www.daria.co.uk** and most sources of QL software. The Zip package is free to download and use and licence terms are clearly explained in the text files which come with it. I think

you even get the C source files if you wish to have a look at how it was programmed. QL zip files and PC zip files are largely compatible, in that if you zip up a set of files on a QL, and copy it to a PC floppy disk, the PC version of Zip can usually read it, although there are one or two QL-specific issues such as storage of QL executable program file headers and dataspace which mean that care and forethought is required if attempting to unzip something on an operating system other than the one on which it was originally zipped. If you are not unzipping QL programs (e.g. unzipping only text files) you may well find that a PC version of Unzip can decode a QL zip file and vice versa.

Zip is not the only such archiving and file compression program of course. There's also BZIP, GZIP, RAR, LHA, ARJ, Zoo and so on. Zip is probably the most common one as far as the QL is concerned, but you may well come across LHA or the LHQ version, and some older QL program archives are in the _Zoo format as well, especially on bulletin board systems. Thierry Godefroy's Archivers Control Panel supports Arc, Lha, Lhq, Zoo and Zip format archives, as long as you have the base programs, which are commonly available from QL PD libraries and some websites.

At the moment, it is not easy to download software from the web direct onto a QL (unless you are using soql or one of the emulators with access to TCP/IP I suppose) so I suppose most of us will be downloading QL software using a PC and transferring the software onto a QL in some form:

- Floppy disk transfer - there are several programs around (such as Discover) which will copy files between QL and PC floppy disks. Such are programs are a bit rarer for Macs, but some Macs are able to read PC disks.
- SMSQ/E users have inbuilt facilities to copy files to and from a PC disk on most QL platforms. Similarly, most emulators will either read files from a QL formatted disk, or a program such as QLTools may be used to copy files to and from a QL disk.
- Jonathan Hudson has packages called QXLTools, WXQLTools and WXQT2 which run on Windows or Unix/Linux platforms and can copy files between the platform on which it runs and QL media.
- Although I have never tried this, it is possible to set up a serial cable between a PC and a QL or Q40, for example and transfer files (usually slowly!) between computers. While serial links are useful, they can be slow (the basic QL is unlikely to manage better than 9,600 baud and serial cables are notoriously fiddly to wire up and get working).
- If you want to get software directly onto a QL from a bulletin board system such as the one operated by TF Services, it should be possible to set up a QL and modem to connect to the bulletin board with software such as QTPI by Jonathan Hudson.

When you unzip (or un Zoo or un-LHA or whatever) software packages, you need to look for files with names such as README, README_TXT or README_DOC to look for distribution and copyright information as well as program instructions. README or README_TXT files are usually plain text files which may be read with most QL editors or imported into word processors like Quill. README_DOC files are usually Quill document files, although some authors have been known to use the filename extension DOC for Perfection word processor documents as well as Quill or Xchange ones. In general, a README file is intended to be the first thing you look at!

There are some QL packages which have been ported over from other systems originally, so you may need to look for other files. A common filename is FILE_ID.DIZ, which seems to crop up frequently in DOS program packages originating on bulletin board systems. I don't really know where this weird name originated, but I guess it's probably distribution notes. There's usually 3 or 4 lines of descriptive text and that's it. I've seen one or two QL programs ported from the PC scene and clipart packages in particular including the FILE_ID_DIZ (note QL underscore instead of a PC dot).

After writing this article, I find myself looking forward to the day when I have soql, QL browser, email and possibly even QL ftp and website designer software, but that's a subject for another article!

JOCHEN MERZ SOFTWARE

Im stillen Winkel 12 D-47169 Duisburg
Tel. 0203 502011 Fax 0203 502012
http://smsq.j-m-s.com smsq@j m s.com

J-M-S is moving!

But the only visible change is a new street name! Please note, that the address change will become active on October, 1st., 2004. It will be

Kaiser-Wilhelm-Str. 302

and even the postcode remains the same! It is possible that during the move, phone and internet facilities will stop working for a day or two, but the phone numbers will move with me.

Happy Birthday J-M-S!

J-M-S celebrates its 20th birthday! Yes, time moves on quite fast. 20 years is a very long time in the computer business, so this deserves a very special offer: aimed at everybody who is still using the QL (or cannot use it anymore because it broke down) and would like to upgrade to a faster machine, who misses it on the PC and would like to access QL programs and QL data on disks, we have created a pack consisting of QPC 2 in its current version for Windows 98 through XP, QD 2003 with the new colours, QPAC 2 in the current version and of course FiFi, the FileFinder. Plus the colour utility disk. **This is software worth over EUR 234,-** if purchased separately, and our celebration offer gives you all this **for only EUR 99,90** in the Celebration-Restarter-Pack!! Offer ends January 2005. Plus postage.

You can order via letter, fax, email (SMSQ@J-M-S.COM), or by using our secure order form on

SMSQ.J-M-S.COM

Roulette

Stephen Poole

When my daughter was 6 years old, she and a friend of hers asked me to write a game for them. At the time I thought it would be a good occasion to rid them of their taste for Games of Chance, and so decided to write a roulette program. Thereby they could soon see that the more often you play the more often you lose, even when the dice are not loaded, and that you need to play an enormous amount before you tend to break even under the best of conditions.

That was the theory anyway. To make things easy for them, they could load up the game by resetting the computer and inserting a microdrive cartridge with a boot file on it. To make things even, one of the first instructions in the program was RANDOMISE which theoretically should have shuffled the random number generator seed. However, the children eventually noted that the same series of winning numbers always occurred, and, with this information to hand broke the bank every time.

Even using RANDOMISE DATE made no difference, so I presumed that the random number generator was faulty, which was indeed the case, and I stopped using it for games. Mark Knight eventually published a fix in QL Today which is used in this program, so I am now happy to go ahead and distribute the Roulette game.

Before first playing the game, you must call the 'configure' routine to initialise the high scores. The code is fairly self-

```
100 ::
110 REMark roulette, by S.Poole, v1985, v13jan2004.
120  REMark before your first game, call 'configure'.
130  CLEAR: init: main: STOP
140 :
150  DEFine PROCedure init
160  OPEN#1,con_64: WINDOW 512,256,0,0: CLS:
     WINDOW#1,256,206,256,0: CLS
170  OPEN#2,con_64: WINDOW#2,256,206,0,0: CLS#2
180  OPEN#3,con_64: WINDOW#3,512,48,0,208: CLS#3
190  OVER 0: FILL 0: INK 7: INK#3,7: PAPER#2,0: INK#2,7
200  ngl=0: z=0: r=37: sc=0: CLS#2: CSIZE#3,3,1: CLS#3
210  OPEN_IN#4,flp1_high_scores
220   INPUT#4,high$: INPUT#4,CASHE$
230  CLOSE#4
240  END DEFine
250 :
260 DEFine PROCedure configure
270  OPEN_NEW#4,flp1_high_scores
280   PRINT#4,'nobody': PRINT#4,'0'
290  CLOSE#4
300 END DEFine
310 :
320 DEFine PROCedure main
330  levels: board: numbers: balance: bank=1000
340 REPeat t
350   True_Randomise: REMark Thanks to Mark Knight QLT
      1,6,p.14
360   sc=sc+1: balance: bets: CLS#2: ct=99: game
370 END REPeat t
380 END DEFine
390 :
400 DEFine PROCedure True_Randomise
410  adr=ALCHP(4)
420  IF adr<0 : RANDOMISE: RETurn
450  POKE_L adr,DATE
460  RANDOMISE PEEK_W(adr+2)
470  RECHP adr
480 END DEFine
490 :
500 DEFine PROCedure tray
510  TURNTO#2,270: MOVE#2,35: TURNTO#2,0: MOVE#2,9
520  TURNTO#2,270: MOVE#2,5 : TURNTO#2,0: MOVE#2,-18
530  TURNTO#2,90 : MOVE#2,5 : TURNTO#2,0: MOVE#2,9
540  TURNTO#2,90: MOVE#2,35
550 END DEFine
560 :
570 DEFine PROCedure balance
580  SCALE#2,100,0,0: ngl=(capital/50)-20: CLS#2
590  AT#2,1,0 : CLS#2,3
600  AT#2,1,0 : PRINT#2,'credit'
610  AT#2,1,35: PRINT#2,'bank'
620  AT#2,2,0 : CLS#2,3
630  AT#2,2,0 : PRINT#2,capital
640  AT#2,2,35: PRINT#2,bank
650  BLOCK#2,200,40,25,155,7
660  BLOCK#2,180,25,33,163,2
670  BLOCK#2,10,120,120,35,7
680  CIRCLE#2,45,75,5
690  PENDOWN#2: TURNTO#2,ngl: MOVE#2,35: tray
700  TURNTO#2,ngl: MOVE#2,-35*2: tray
710 END DEFine
720 :
730 DEFine PROCedure board
740  FILL 1: INK 0: CIRCLE 0,0,47: FILL 0
750  FILL 1: INK 4: CIRCLE 0,0,45: FILL 0
760  FILL 1: INK 0: CIRCLE 0,0,30: FILL 0
770  FILL 1: INK 2: CIRCLE 0,0,27: FILL 0: PENUP
780 FOR f=steps TO 360 STEP steps
790     TURNTO f: MOVE 30: PENDOWN: MOVE 15 : PENUP
800     MOVE -30:    INK 0: PENDOWN: MOVE -15: PENUP
```

```
810 END FOR f
820 END DEFine
830 :
840 DEFine PROCedure levels
850  CSIZE#2,3,1: PRINT#2,' ROULETTE': CSIZE#2,0,0
860  CLS#3: CSIZE#3,0,0: BEEP 12345,23
870  PRINT#3,\\\,'DIFFICULTY...1(easy),2,3(hard) or [Q]uit?'
880  q=CODE(INKEY$(#1,-1))
890 SELect ON q
900     =CODE('1'): steps=90
910     =CODE('2'): steps=60
920     =CODE('3'): steps=15
930     =CODE('q'),CODE('Q'): STOP
940     =REMAINDER : GO TO 870
950 END SELect : capital=1000
960 SCALE 100,-45,-50: CIRCLE 0,0,1
970 END DEFine
980 :
990 DEFine PROCedure numbers
1000 FOR f= steps TO 360 STEP steps
1010      x=r*COS(RAD(f)): y=r*SIN(RAD(f))
1020      FILL 1: INK 0: CIRCLE x,y,2: FILL 0
1030      INK 7: CURSOR x,y,0,0: PRINT f
1040 END FOR f
1050 END DEFine
1060 :
1070 DEFine PROCedure bets
1080 BEEP 12345,33: INK 7
1090 IF capital<=0 THEN
1100    ngl=ngl-3: balance
1110    CLS#3: PRINT#3,'You are BANKRUPT..!!!': BEEP 12345,255
1120    PRINT#3,'Another Game? (y/n)': IF INKEY$(#1,999)=='y':
        RUN: ELSE STOP
1130 END IF
1140 ::
1150 CLS#3: INPUT#3,'Which Hole..?'!z$
1160 IF z$<steps OR z$>'360': GO TO 1150
1170 IF z$ MOD steps: GO TO 1150
1180 BEEP 12345,33: INPUT#3,'How much are you betting?...'!bet
1190 IF bet>capital: GO TO 1180
1200 capital=capital-bet: bank=bank+bet
1210 FOR f=steps TO 360 STEP steps
1220      IF f=z$: z=1: END IF
1230 END FOR f
1240 IF z<>1: GO TO 1150
1250 END DEFine
1260 :
1270 DEFine PROCedure game
1280  OVER 0: FILL 0: INK 7
1290 FOR j=1 TO 4
1300      IF j=3: ctt=360/steps: ct=RND(1 TO ctt)
1310      FOR f= steps TO 360 STEP steps
1320          x=r*COS(RAD(f)): y=r*SIN(RAD(f))
1330          ct=ct-1: OVER -1: FILL 1: CIRCLE x,y,2: BEEP 999,9
1340          IF ct=-1: BEEP 12345,99: i$=INKEY$(#1,99): EXIT j
1350          OVER -1: FILL 1: CIRCLE x,y,2: FILL 0: OVER 0
1360      END FOR f
1370 END FOR j
1380  IF ct=-1: OVER -1: FILL 1: CIRCLE x,y,2: OVER 0: FILL 0
1390  IF z$=f THEN
1400      gains=bet*(360/steps): REMark adapt wins as required.
1410      bank=bank-gains
1420      capital=capital+gains
1430      CLS#3: PRINT#3,'Well Done..!! After'!sc!'tries, you have won'!gains
1440      sc=0: BEEP 12345,0: i$=INKEY$(#1,222)
1450      IF bank<=0 THEN
1460         balance: CLS#3: PRINT#3,'BRAVO!! You broke the BANK..!!'
1470         i$=INKEY$(#1,99): files: RUN
1480      END IF
1490      RETurn
1500 END IF
```

explanatory, so I have included few REMarks. The program will LRUN under QDOS or EXEC under SMSQ/E. The first page prompts for a difficulty factor to be input, where the more holes there are, the longer you have to wait for winnings. Then on the left a bank balance is drawn to help younger children see how they are faring. On the right a roulette board is drawn with numbered holes allowing selection to be input. Input is error-trapped.

This scheme is quite intuitive, (especially for younger children), and various messages appear on screen as the game progresses. Our children played happily for hours with the game, (that is until they discovered how to win, when they soon got bored with it!). You could modify the winnings to suit yourselves, but as things are they are set for break-even. Of course neither QL Today nor I can accept any responsability for any mis-use you may make of the program. The code could be adapted to graphically simulate a real Roulette Wheel, but that would complicate the listing, which is designed for simplicity.

Good Luck!

```
1510 CLS#3: BEEP 12345,255: PRINT#3,'You Lost': i$=INKEY$(#1,99): CLS#3
1520 END DEFine
1530 :
1540 DEFine PROCedure files
1550 b 0: b 7: b 7: b 7: b 0: b 9: b 9: b 1: b 2: b 3: b 4
1560 b 3: b 2: b 1: b 5: b 5: b 5: b 6: b 7: b 8: b 9: b 7: b 9: b 9
1570 CLS: INPUT 'Inscribe your name..'!best$!!: PRINT capital
1580 IF capital>CASHE$ THEN
1590    PRINT'You have beaten'\high$!CASHE$: cash$=capital: i$=INKEY$(#1,333)
1600    high$=best$: CASHE$=cash$
1610    DELETE flp1_high_scores
1620    OPEN_NEW#4,flp1_high_scores
1630    PRINT#4,high$: PRINT#4,cash$: CLOSE#4: RETurn
1640 END IF
1650 CLS: PRINT best$!capital\"..hasn't beaten"!high$!CASHE$: i$=INKEY$(#1,333)
1660 END DEFine
1670 :
1680 DEFine PROCedure b(bp)
1690 BEEP 999,bp*10: i$=INKEY$(#1,9)
1700 END DEFine
1710 ::
```

Using the new WMAN Colours from SBASIC

by Dilwyn Jones

We have already published articles about the new Window Manager - in particular, this article draws upon information from articles by Wolfgang Lenerz in Volume 7 Issue 8 and Volume 8 Issue 1.
Although those articles were strong on the theory, they did not contain many examples of how to use the colours.

Listing 1 is a short SBASIC program showing how to use the new Window Manager SBASIC extensions to draw a simple program display which looks like the "standard" appearance of pointer driven programs such as the QPAC2 menus, software from Jochen Merz and so on. Marcel Kilgus, who wrote the new Window Manager, has now added some new SBASIC keyword which make it easier to access the Window Manager palettes from BASIC.
The idea behind the Window Manager colour palettes is to provide a mechanism whereby programs can use standard colour schemes. The original QDOS mode 4 environment allowed for 4 schemes as follows:

0 = white paper, green and white stripes across the black ink title
1 = black paper, red and black stripes across the white ink title
2 = white paper, red and white stripes across the black ink title
3 = black paper, green and black stripes across the white ink title

I have somewhat simplified those descriptions, as other elements such as loose items, information windows and so on can all have their colours defined in this way, I'm just going to use the above settings to provide a simplified explanation of what's going on.

To be able to understand this article, you will need some knowledge of the terminology used by authors of pointer driven programs. You need to know about information windows, loose items, application windows and so on. If you are not familir with these terms, I suggest you get hold of a copy of Norman Dunbar's excellent Pointer Environment Idiot's Guide, which is available from PD libraries and other sources of good, free QL software. Some knowledge of hexadecimal (base 16) numbers will also be helpful, as it is easier to deal with these new colour values in hexadecimal or byte sized units rather than trying to convert 16 bit values to unhelpful decimal numbers.

To use these colours, we use the new commands which have the WM_ prefix. These are:

```
WM_PAPER [#channel],colour

WM_STRIP [#channel],colour

WM_INK [#channel],colour

WM_BORDER [#channel],colour

WM_BLOCK [#channel],colour
```

These work in a similar way to the equivalent PAPER, STRIP, INK, BORDER and BLOCK commands in SBASIC, but these new commands use the Window Manager's colour palettes.

The basic principle is that the Window Manager maintains a list of standard colour schemes. What this means is that there's an itemised list of colours. In fact, there's seven types of lists. We will be concentrating on the System Palette, but there's no harm in reminding ourselves what they all are:

1. As before, simple colour values 0–255 in the second byte, the first byte being 0. In hexadecimal, this is represented as $00xx. The $00 signifies the original colour values and the xx is the colour number from 0 to 255.

2. The colour palettes. Values starting with $01 (decimal 1) in the first byte, followed by a number from 0 to 255 to indicate colour number.

3. System Palette, the one we are interested in. Indicated by a value of $02 (decimal 2) in the first byte, and a value from 0 to 255 in the second byte. In this list, each of the possible 256 values has a specific purpose, for example, the first byte in the list tells us the colour number for the window border.

4. Grey scale. A list of up to 256 different shades of grey to use. The first byte has a value of $03 (decimal 3) and the second byte has a value of 0 to 255.

5. Border colours. This one is used for border colours only, which we'll go into in more detail at th end of this article with a little program to investigate the new borders. The first byte has a value of $04 (decimal 4), with the second byte indicating details such as whether the 3D border is raised or lowered, the border type and a compatibility mode.

6. Palette stipples. With these colours, you can have a stipple of two 6-bit colour numbers. This type is indicated by the most significant two bits of the first byte having a value of binary %01, the next two bits give the stipple number, the next 6 bits give the stipple colour and the next 6 bits the main colour.

7. A 15 bit RGB value is indicated by the most significant bit of the first byte being set. The next 5 bits give the red component, the next 5 bits the green component, the next 5 bits the blue component.

You don't really need to understand the above as yet. All you need to know is that we'll be making use of the colours recorded in list 3. above (System Palette). To see what each item in this palette refers to, see Wolfgang's list on page 50 of Volume 8 Issue 1 of QL Today. At the moment, there's only the old types 0 to 3 built in, but Wolfgang did explain how to set up your own lists of colours to create your own colour schemes, which is what the System Palette is all about.

A Palette is basically like an artist's paint palette. Imagine that it has a certain number of colours and that the artist has several such palettes to hand. He can put the same colours in a different order in each of his palettes. So, on a normal day he has green (for the grass) in the first paint slot, blue (for the sky) in the second slot, white (for the clouds) in the third and yellow (for the sun) in the fourth.
One day, someone asks him to paint everything in a different colour, "just to be different". So as he's a creature of habit, he paints everything in the same order, so he grabs a fresh palette to do his painting with. He wants the grass to be blue, so he puts blue in the first slot, the sun is green, so puts the rgeen paint in the second slot, the clouds need to be yellow, so the third slot contains yellow paint, the sun has to be white, so the white paint goes in the fourth slot.

In other words, the list is determined as follows:

Colour 0 (trees colour for this customer)
Colour 1 (sky colour for this customer)
Colour 2 (cloud colour for this customer)
Colour 3 (sun colour for this customer)

So he files these away and always selects the appropriate labelled palette for his customers when they request a particular colour scheme.

OK, so it wasn't a good real life example, but the comparison is there. Hopefully, you will see that the idea is to provide a number of colour schemes, so that all programs which use the same colour scheme will look alike and generally be consistent in appearance.

The difference between the new WM_xxx commands and the older equivalents is that the new commands look up in the Window Manager co-

lour tables. The colour values used for the new ones may be different to the older commands but they are in principle the same.
The older commands cannot access the colour scheme mechanisms of the Window Manager. Carry on using the old commands if you are not concerned about standard appearances and Window Manager colour schemes, or use the new WM_xxx commands if you wish to use the Window Manager colours.

Run the short program in listing one on a recent version of SMSQ/E and SBASIC to see how the 4 built in colour schemes look, and how the new keywords access the necessary information. You may like to use the DISP_COLOUR command to see the effect in the different colour modes, e.g. DISP_COLOUR 0 to set QL colours mode, DISP_COLOUR 3 for 16-bit colour on QPC2, QXL or Q40/Q60, or DISP_COLOUR 2 for 256 colour mode on Aurora or QPC2.

The program starts by asking you which of the four colour schemes you'd like to examine. Enter a number from 0 to 3. Line 120 just sets some standard colours (white ink on black paper) to see that the INPUT prompt can be read if you have already scrambled up the colours! I've used screen window #1 throughout for the example.

Line 140 selects which palette number to use. The SP_JOBPAL command selects which colour scheme to use for the specified JOB ID. In common with many operating system related commands, the job ID number can be -1 to indicate 'current job' or 'myself'. SP_JOBPAL -1,0 for example means 'set my own system palette to number 0'. This command lets us completely change the program's colour scheme without having to change the colour values of all INK, PAPER, BORDER, STRIP and BLOCK commands, simply by stating which colour scheme to use!

From there on, we simply access the same entry in the list for whichever of the available palettes we selected. Can you see the comparison with the artist's palettes described above?

We now have to use the list described by Wolfgang on page 50 of volume 8 issue 1 to see which number refers to which element of the display. For the purposes of this example program, we'll use the following elements:

0 – window border
1 – window background (paper)
2 – window foreground (ink)
4 – title background at the top of the window (a strip of paper)
5 – title text background, the strip colour behind the ink of the title
6 – title foreground, the ink number used for the title text

These are used in a series of WM_PAPER, WM_INK and WM_STRIP commands to set the appropriate colours for the various parts of the screen.
Line 150 uses a WM_BORDER command to set the border colour to the colour indicated by entry 0 in the table. Entry 0 always indicates the colour to be used for the main window border.
Line 160 uses a WM_PAPER command to setthe window background to the colour indicated by entry 1 in the table.

Note how it is easier to use hexadecimal numbers to indicate what you are trying to do. In the case of the WM_PAPER command in line 160, the colour number is given as hex 0201 (or $0201 since SBASIC allows us to use the '$' symbol to precede a hex number). This shows us that we are using colour format 2 (System Palette), entry 1 (window background colour). It's hard going at first, but write out a few examples and you'll soon get the hang of it!
Having set the paper colour to window manager colour $0201, we use CLS to clear the screen to that paper colour. Hopefully, you can now see that WM_PAPER and PAPER are very similar, they just get the colour numbers from different systems.

Line 170 sets the main ink colour for the window to the value specified as colour $0202 (window foreground colour). At present, I haven't made the program print anything in this colour, but if you want to see it, you could add a line:

```
175 AT#1,3,0:PRINT'Foreground colour':AT#1,0,0
```

Next, we are going to create a one line title along the top of window #1. To do this, we'll tell the program to create a line of title background colour (number $0204) and in the middle (approximately!) we'll put a little title with its own background colour (see figure 1 for a screen dump of how it should look!).

Line 190 sets the window manager paper colour for the line across the top of the window with a WM_PAPER command, then a CLS #1,3

RWAP Services NEWS!

New Websites!

We are proud to present our new websites! Starting September, we have separated our business in three separate websites.

You can find them at:

<http://www.rwapadventures.com/> (Adventure Games)
<http://www.rwapservices.co.uk/> (General Site) and
<Http://www.rwapsoftware.co.uk/> (Software)

Our old address: <http://hometown.aol.co.uk/RWAPSoftware/> is of course still functioning but will be deprecated in the future.

New Products!

QWORD 1.0 NOW WITH DIGITAL SOUND ON QPC2!

The wait is now over! Q-Word version 1 is finally available!

Platforms:
QPC/QXL, Q40/Q60, Aurora (with SGC)

Prices:

All versions without P-Word	£20.00p
All versions with P-Word	£30.00p

Notes:
Q-Word DOES NOT require SMSQ/E with GD2 support -OR- SMSQ/E at all on the Aurora or Qx0 machines. It works on the highest colour depth everywhere regardless of Operating System.
The Aurora version is available on either HD or ED disk. For the latter add £1.00 to the price. ED version is uncompressed and can be run directly from the floppy. All other Floppy versions are compressed. QPC/QXL version comes on CD. Non CD versions DO NOW support digital sound on QPC2

Quantum Leap ED Drives

After many years of unavailability, here they are again! These are high quality Mitsubishi constructed/IBM badged drives with full warranty.
Unlike previously sold ED drives, these do not require FLP_JIGGLE and have no problems formatting 720K disks. However for the latter a switch is included with the cable.

ED Bare unit (no cable)	£ 29.99p
ED Boxed unit (complete with cable/ PSU)	£ 98.99p
Single unit Cable (with switch)	£ 4.99p
Dual unit Cable (with two switches)	£ 5.99p

(More options available, contact us for details. Also available mass quantities of ED and DD disks!)

TALENT COMPUTER SYSTEMS *for* **Windows**

For QLers that run Windows or with incompatible hardware for Talent Games, we now have re-released these adventures so that they can run on your Windows-equipped PC. No Emulator, floppies, microdrive backups etc. required, just a one-click install! Of course the full QL line is still available! (See side column)

Talent Games for Windows	ea. £ 10.00p

(Each Game includes a runtime installation of QLAY-2 by Jimmy Montesinos)

Old Favourites!

Utilities

Sidewriter v1.08	£ 10.00p
Landscape Printing (EPSON printers)	
ImageD v1.03	£ 10.00p
3D object generator	
Q-Help v1.06	£ 10.00p
Superbasic On-Screen help system	
Q-Index v1.05	£ 5.00p
Keyword-to-topic finder	
ProForma ESC/P2 Drivers v1.04 for ProWeSs	£ 8.00p
Printer Driver	

Applications

Flashback SE v2.03 (upgrade only)	£ 5.00p
Database	
QL Cash Trader v3.7	£ 5.00p
Accounting/Finance	
QL Payroll v3.5	£ 5.00p
Accounting/Finance	
QL Genealogist v3.26	£ 20.00p
Genealogy	
Genealogy for Windows	£ 50.00p
QL Genealogist to Windows version upgrade	£ 25.00p
QL Cosmos v2.04	£ 5.00p
Planetarium	
Q-Route v2.00	£ 25.00p
Route Finding	
Upgrade from v1.xx	£ 5.00p
Britain map v1.11	£ 2.00p
BIG Britain map (needs 2Mb) v2.03	£ 5.00p
Various Britain Area maps (ask for details)	ea. £ 2.00p
Ireland map v1.00	£ 5.00p
Belgium map v1.01	£ 2.00p
Catalonia map v1.03	£ 2.00p
P-Word UK English Dictionary (500.000 words!)	£ 15.00p
Dictionary	

Leisure

Return to Eden v3.08	£ 10.00p
Adventure	
Nemesis MkII v2.03	£ 8.00p
Adventure	
The Prawn v2.01	£ 8.00p
Adventure	
Horrorday v3.1	£ 8.00p
Adventure	
West v2.00	£ 5.00p
Adventure	
The Lost Kingdom of Zkul v2.01	£ 5.00p
Adventure	
All 6 games above	£ 25.00p
D-Day MkII v3.04	£ 10.00p
Strategy/War Simulation	
Grey Wolf v1.08	£ 8.00p
Graphical Submarine Simulation	
War in the East MkII v1.24 (upgrade only)	£ 5.00p
Strategy/War Simulation	
Open Golf v5.20	£ 8.00p
Sports Simulation	
QuizMaster II v2.07	£ 5.00p
Quiz	
Stone Raider II v2.00	£ 5.00p
Arcade Game	
Hoverzone v1.2	£ 5.00p
Arcade Game	
Deathstrike v1.5	£ 5.00p
Arcade Game	
Flightdeck v1.0	£ 10.00p
Flight Simulation	
All 6 games above (Open Golf, QuizMaster II,Stone Raider II, Hoverzone, Deathstrike and Flightdeck)	£ 28.00p

Notes on Software requirements
The following programs have a minimum SGC card requirement: P-Word, Qword, Big Britain MAP for Q-Route

RWAP Services
35 Chantry Croft, Kinsley, Pontefract, West Yorkshire, WF9 5JH, United Kingdom Tel: +44 1977 610509 (From the UK dial: 01977 610509)
Website: http://www.rwapservices.co.uk

We accept:

PayPal VERIFIED NOCHEX

(For PayPal please add 4% to the total price. Cheques in £ sterling made payable to R. Mellor)

command is used to clear one text line to the stripey colours used for title lines by the window manager.

Line 200 sets the ink colour for the little title part in the middle of this line. The WM_INK is set to entry $0206 in the currently selected palette.

This title text needs to be on a differently coloured background strip, so we use WM_STRIP to set the background to entry $0205 in the currently selected palette.

Finally, line 210 positions the text to be printed, so you get a window in standard colours with a short heading of "TITLE TEXT" at the top.

When you run the program, you get to see what the 'standard' colours are for the 4 inbuilt system palettes. You can create extra palettes of your own, as Wolfgang described, but this is a bit too advanced for this article. What I would suggest you do next is to investigate the other values in the System Palette. Make sure you know what an information window is (a window used to just display information without any menus or selectable items in it), what an application window is (usually where the action takes place, where you select something from a menu or list, for example), what a loose item is (an item you can select by clicking a mouse button when the pointer is over it, for example, some programs have a little [X] or ESC symbol which tells the program to stop when you select it) and so on. Some of the items in Wolfgang's list are not very obvious as to what they are, but you'll pretty quickly get the general idea.

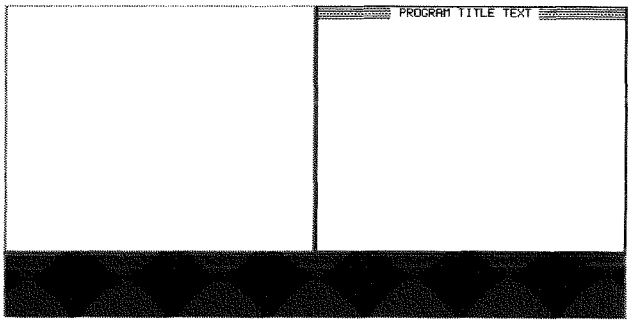


Figure 1. palette.gif "What the display produced by Listing 1 should look like!

Borders

The new Window Manager includes all sorts of extended borders, including some 3D-style ones. Listing 2 is a little program to help you see how it all works and what they look like. In fact, it goes a stage further by allowing you to view borders in 5 of the available colour systems, as described in lines 120 to 160. The number entered in line 170 corresponds to the first byte of the colour format – $00 to $04 in hex, or 0 to 4 in decimal of course. As this needs to be given as the top byte of a 16-bit value in the WM_BORDER command, we multiply this by 256 in line 200. Line 210 sets up a loop to cycle through all 256 possible colour values. Again, you may wish to experiment with different colour modes set up with DISP_COLOUR as described above to see how it works in QL colour modes, 16 bit colour modes and so on.

Line 220 sets the border colour with the WM_BORDER command. As it stands, it's a border width of 2 but you may like to alter this to a different width to study how the different 3D border effects work. Line 230 displays the binary

```
100 REMark create a little program display in 'system' palette colours
110 COLOUR_PAL : REMark ensure we use palette colours
120 PAPER #1,0 : CLS #1 : INK #1,7 : REMark ensure we can see the INPUT!
130 INPUT 'Palette 0-3 ? ';palet
140 SP_JOBPAL -1,palet : REMark select system palette number for current job
150 WM_BORDER #1,1,$0200 : REMark main window border
160 WM_PAPER #1,$0201 : CLS #1 : REMark main window background
170 WM_INK #1,$0202 : REMark main window foreground
180 REMark create title strip
190 WM_PAPER #1,$0204 : CLS #1,3 : REMark top title strip
200 WM_INK #1,$0206 : REMark title foreground
210 WM_STRIP #1,$0205 : REMark title text strip colour
220 AT 0,10:PRINT' PROGRAM TITLE TEXT '
```

Listing 1. Create a standard appearance display in channel #1

equivalent of the colour number so that you can see how it all ties up with what Wolfgang explained on page 47 of Volume 8 Issue 1. You could also display the hex value if you wish, by altering line 230 to:

```
230  AT #0,1,0 : PRINT #0,a TO 10;
     BIN$(scheme+a,16)!HEX$(scheme+a,16)
```

Between each colour number in the loop, there is a PAUSE statement at line 240 which makes the program wait until you press a key before displaying the next colour

```
100 REMark test new window manager colour schemes
110 PAPER #1,0 : CLS : CLS #0
120 PRINT'0 - Old colours'
130 PRINT'1 - Palette'
140 PRINT'2 - System palette'
150 PRINT'3 - Grey scale'
160 PRINT'4 - 3D borders'
170 INPUT\'Select scheme 0-4 ';scheme
180 CLS : CLS #0
190 PRINT #0,'Showing scheme ';scheme
200 scheme = 256*scheme
210 FOR a = 0 TO 255
220  WM_BORDER #1,2,scheme+a
230  AT #0,1,0 : PRINT #0,a TO 10;BIN$(scheme+a,16)
240  PAUSE
250 END FOR a
```

Listing 2. View the new borders

To see the border colours built into the 3 standard colour schemes you should really add line 115 like this:

```
115 INPUT"Palette 0-3?";palet:
    SP_JOBPAL -1,palet
```

which will ask you which of System Palettes 0 to 3 to use. This way, in QL colour mode, for example, you'll find how it uses red and white, green and red and so on for the 3D effects and shapes. In fact, going back to the first listing, where we discussed the system palette colour list, the colours used for the 'dark' and 'light' parts of the border are stored at locations $30 and $31 (decimal 48 and 49) of the palette entry numbers. So when you are moving to the more advanced feature of defining your own colour palettes, you can specify your own pair of border colours for your system palette.

I hope that by now it's clear what all this is about. As I hinted above, System Palettes are a really brilliant way of changing the appearance of a program simply by telling it which list of colours to use! You won't need to change all the INK, PAPER, BORDER etc colour numbers in each of these statements in your program. Simply change the system palette instead!

I hope this article has been useful in explaining how to make use of the System Palette and the new SBASIC keywords. If you have any more example listings or articles on this subject, I would be very grateful to receive them for publication.

Installing and Using UQLX and UQLX-win32

Timothy Swenson

Recently I was able to pick up a 200 Mhz Pentium II system that was being disposed of at work. I decided that this would make a good system to put Linux on and to try UQLX. I've used UQLX before, but it's been about 4 years. I wanted to see what is new with UQLX and see how well it could do. Since I was tinkering with regular UQLX, I decided to try out the Windows port of UQLX called UQLX-win32.

Ideally I'm looking to see how close UQLX can come to working like a full QL system (be it a Gold Card QL or a Q40). I had some problems in the past with UQLX in that it used the native Unix file system. The Unix file system is case sensitive where as the QDOS file sytem is not. Well, QDOS displays different cases, but resolution is case insensitive. So, the file BOB_txt is the same as bob_txt in QDOS, but in Unix, they are different. The latest UQLX documenation mentions that it now supports QXL.WIN files, so this gives a native QDOS file system.

UQLX

I loaded Red Hat 8.0 on the system. Red Hat allows you to load different packages on the system based on what you are going to use it for. I chose the Workstation or Desktop option. That was a bad idea. The C compiler, GCC, is not loaded with this option and I had to load it manually, plus some other library packages. Since you have to compile UQLX, I'd recommend choosing the Developer installation.

Installing UQLX

Once I had Linux up and running, the next step was to get UQLX. UQLX can be found at the following address:

http://linux-q40.sourceforge.net/uqlx/

Download the file uqlx.tar.bz2. I put the file in my local home directory and noticed that it was called uqlx.tar.tar. So I renamed it to uqlx.tar.bz2, unzipped it and untarred it using the following commands:

```
% mv uqlx.tar.tar uqlx.tar.bz2
% bunzip2 uqlx.tar.bz2
% tar -xvf uqlx.tar
```

This will create a 'uqlx' directory. Doing the compiling is fairly simple under Linux:

```
% cd uqlx
% ./MK.all
% make install
```

For me, the compile went smoothly (after I had all the necessary packages loaded) and soon I had the 'qm' binary ready to run.

Configuring UQLX

UQLX has a .uqlxrc configuration file that resides in your home directory (and not in the uqlx directory). You use this to define your UQLX environment, including ROM, local drives, floppy drive, etc. If you run UQLX before configuring this file, a basic .uqlxrc file will be created for you and used. At this point, you can then just edit the file as you find necessary. The default version of the file gives some fairly good examples of what options you have. The UQLX documentation goes into some good detail on the options, what they are, and how to use them.

UQLX comes with two QL roms, the JSU and an early version of Minerva (1.89). The .uqlxrc file defines an entry for a ToolKit II rom, but you need to supply this yourself.

File Systems

UQLX supports two disk systems, the native Unix file system and a QDOS file system image. By default when defining a drive (MDV, RAM, FLP or WIN), the native Unix file system is used, including Unix name resolution. Files are stored as normal Unix files and all of the header information is stored in a special '.-UQLX-' file. If you use the 'qdos-like' option, then name resolution is more QDOS like. When using the native Unix file system, be careful to not manipulate files via the shell as this will not change anything in the '.-UQLX-' file and can really mess up the file system points for QDOS.

The other file system type is 'qdos-fs'. This means a full QDOS file system that is really one large Unix file. This can be either a disk-image or a QXL.WIN file. One way to get a disk image is to put a QDOS formatted disk in the floppy drive and use 'dd' to copy the entire 720K or 1.44M disk to a single file. A QXL.WIN file is another matter.

UQLX does not come with any tools to create or manage QXL.WIN files. After some thinking, I realised that Jonathan Hudson wrote a program called qxl_tool that creates and manages QXL.WIN files for both Windows and Linux. I downloaded the tool at Jonathan's from web page:

http://www.daria.co.uk/

Using the instructions that came with qxl_tool, I compiled the application on my Linux system. Now I was able to create an empty QXL.WIN file that would be my WIN1_ drive. The file itself can be called whatever you want and it does not have to end in .WIN. In the .uqlxrc file the line to define WIN1_ is:

```
DEVICE = WIN1, ~/qxl_WIN1.win, qdos-fs
```

now I have a huge 300MB qxl_WIN1.win file sitting in my home directory.

By default when running UQLX with Linux, FLP1_ is defined as the local physical floppy drive (/mnt/floppy). So, the command DIR FLP1_, will directory a QDOS formatted floppy disk in the floppy drive. You don't need to worry about Linux actually mounting and unmounting the floppy and you can exchange floppies fairly freely.

UQLX requires that an MDV1_ device exist and it will only look for the BOOT program on this device. I tried removing the definition for MDV1_ to see if it will check WIN1_ for the BOOT program, but no luck. I have put the BOOT program on MDV1_ and everything else is on WIN1_, including all of the files called by the BOOT program.

TF Services

Compswitch

A UK 4-way trailing socket designed to switch off computer peripherals automatically when the computer is switched off, or (in the case of an ATX computer) when it auto-powers down. *Compswitch* has one control socket, and three switched sockets. Can be used with lights/hifi/monitors—ie a QL monitor can be used as a switch control.

Cost £24

superHermes

A major hardware upgrade for the QL

All Hermes features (working ser1/2 at 19200, independent baud rates/de-bounced keyboard/ keyclick) IBM AT kbd I/F // HIGH SPEED RS232 at 57600// serial mouse port and 2 other RS232 inputs// 3 I/O lines // EEPROM

Cost (including manual/software) .....**£90** (£92/£93)
IBM AT UK layout Keyboard.....**£11** (£13/£15)
Serial mouse..... **£8** (£8.50/£9)
Capslock/scrollock LED .....**£1** (£1.50/£1.50)
Keyboard or mouse lead .....**£3** (£3.50/£3.50)
High speed serial (ser3) lead.....**£4** (£4.50/£4.50)

Hermes available for £25 (£26/£27) Working ser1/2 and independent input, debounced keyboard.

SuperHermes LITE: All Hermes features (see above) + an IBM AT keyboard interface only.
Cost (incl keyboard lead) .....**£53** (£54/£55)

QL REPAIRS (UK only)

Fixed price for unmodified QLs, excl microdrives. QLs tested with Thorn-EMI rig and ROM software.

£27 incl 6 month guarantee

Minerva

The ORIGINAL system operating system upgrade

OTHER FEATURES COMMON TO ALL VERSIONS

DEBUGGED operating system/ autoboot on reset of power failure/ Multiple Basic/ faster scheduler- graphics (within 10% of lightning) - string handling/ WHEN ERROR/ 2nd screen/ TRACE/ non-English keyboard drivers/ "warm" fast reset. V1.97 with split OUTPUT baud rates (+ Hermes) & built in Multibasic.

First upgrade free. Otherwise send £3 (+£5 for manual if requd). Send disk plus SAE or two IRCs

MKI..£40 (£41/£43) **MKII..£65** (£66/£67)

MINERVA RTC (MKII) + battery for 256 bytes ram. CRASHPROOF clock & I^2C bus for interfacing. Can autoboot from battery backed ram. Quick start-up.

QL RomDisq

Up to 8 mbyte of flash memory for the QL

A small plug in circuit for the QL's ROM port (or Aurora) giving 2, 4 or 8 mbytes of permanent storage - it can be thought of as a portable hard disk on a card, and reads at some 2 mbytes per second.

Think of it - you could fully boot an expanded QL, including all drivers/SMSQ etc off **RomDisq** at hard disk speed with only a memory expansion needed.

2 mbytes RomDisq.....**£39** (£40/£41)
4mbytes RomDisq.....**£65**(£66/£67)
8 mbytes RomDisq.....**£98** (£99/£100)
Aurora adaptor.....**£3** (£3.50/£4)

MPLANE

A low profile powered backplane with ROM port

A three expansion backplane with ROM port included for RomDisq etc. Aurora can be fitted in notebook case and powered off single 5V rail - contact QBranch for details. Two boards (eg Aurora and Gold Card/Super Gold Card/Goldfire fixed to base. Suitable for Aurora (ROM accessible from outside) & QL motherboard in tower case. Specify ROM facing IN towards boards, or OUT towards back of case.

Cost .....**£34** (£35/£36)

I2C INTERFACES

Connects to Minerva MKII and any Philips I^2C bus

Power Driver Interface 16 I/O lines with 12 of these used to control 8 current carrying outputs (source and sink capable)
2 amp (for 8 relays, small motors).....**£40** (£43/£44)
4 amp total (for motors etc).....**£45** (£48/£50)
Relays (8 3a 12v 2-way mains relays (needs 2a power driver).....**£25** (£28/£29)
Parallel Interface Gives 16 input/output lines. Can be used wherever logic signals are required.....**£25** (£27/£28)
Analogue Interface Gives eight 8 bit analogue to digital inputs (ADC) and two 8 bit digital to analogue outputs (DAC). Used for temp measurements, sound sampling (to 5 KHz), x/y plotting.....**£30** (£31/£32)
Temp probe (-40°C to +125°C).....**£10** (£10.50/£11)
Connector for four temp probes.....**£10** (£10.50/£11)
Data sheets.....**£2** (£2.50/£3)
Control software & manual (for all I/F).....**£2** (£2.50/£3)

QL SPARES

Keyboard membrane ..... **no longer on sale**
1377 PAL.....**£3** (£3.50/£4)
Circuit diagrams.....**£3** (£3.50/£4)
68008 cpu or 8049 IPC.....**£8** (£8.50/£9)
8301/8302 or JM ROM or serial lead.....**£10** (£10.50/£11)
Power supply (sea mail overseas).....**£12** (£19/£23)
Other components (sockets etc) also available

Prices include postage and packing (Airmail where applicable) Prices are: UK (Europe /Rest of world). Payment by cheque drawn on bank with UK address,/postal order or CASH! I can no longer accept card payments as UK only does PDQ transaction. SAE or IRC for full list and details 22 Feb 03

29 Longfield Road, TRING, Herts, HP23 4DG
Tel: +44 (0) 1442-828254 Fax/BBS: +44 (0) 1442-828255
tony@firshman.co.uk http://www.firshman.co.uk

Toolkit II does not support level 2 drivers, meaning that I can't use real QDOS directories. I believe these came with the Gold Card and Super Gold Card (and SMSQ/E). If you try to access a floppy or QXL.WIN file with real directories, UQLX will have a problem reading the files. It can list then, but you can't copy or execute them. So, I have to fall back on the older standard of using long filenames and using the DDOWN, DUP, PROG_USE and DATA_USE commands to organize my files. A little bit different from my Gold Card QL, but nothing that can't be worked around. I'll just keep the executables on WIN1_ and put the data on WIN2_.

I'm thinking of creating a WIN3_ that is a "qdos-like" native Unix file system device. One advantage of doing this is that any files I put on this device I can see and use in Unix. If I need to ftp a zip file or text file to another system, I can move it to WIN3_ and then (via a shell) ftp the file to another server. It will make it easier if I need to have Unix print a text file for me.

BOOT

As mentioned above, UQLX expects to see a BOOT program on MDV1_. I took the BOOT program from my Gold Card system and copied it and the boot programs to WIN1_. I then copied the BOOT to MDV1_.
There were a few things that I did not need in the BOOT program. Anything hardware related can be removed, such as Hermes IPC code. UQLX has a built-in mouse driver, so I do not need to load the Serial Mouse driver.

Large Screens

UQLX supports larger screens as long as your X-windows display is larger than 512x256. I've got my X-windows set to 800x600. Large Screens is supported on UQLX only with the Minerva ROM. To start UQLX with a larger screen, here is the command:

```
% qm -g 720x500
```

This is about the largest screen size that I can use on my system. You can also run the following commands to get default larger screens:

```
% qx % qxx % qxxx
```

The size of the main QL windows will not change, including #0. When I executed Xchange, I noticed that it centered itself in the main window, so it seems to handle the larger screens just fine. QPAC II also handles the larger screen size. I opened some QPAC II PE windows and moved them around the larger screen with no problems.

UQLX-win32

UQLX-win32 is available on Pheobus Dokos' web page (**www.dokos-gr.net**). It is contained in a RAR file. By doing a google search on RAR I found a freeware tool that will uncompress a RAR file. I downloaded the UQLX-win32 file to my Windows 2000 laptop and expanded it onto a UQLX directory.

Installing CYGWIN

Because UQLX-win32 relies on the X-windows interface and library, you have to install a port of X-Windows, called CYGWIN, on your MS Windows system. CYGWIN can be found at **www.cygwin.com**. The only parts you need are listed as "Base" and "X11". The simplest way to download and install these parts is to run the "setup.exe" from the CYGWIN site. You can find this on the web page with the words "install now" or the like. Once setup.exe downloads and runs, it lists all of the different packages of CYGWIN. You should see the words "base" and then a funny symbol and then "default". Click on the word "Default" until it becomes "Install". Scroll down and do the same for the "X11" package. You have now selected the "Base" and "X11" packages, and only those packages, to download and install. You then click the "Next" button and continue throught the install routine. I downloaded the packages from my Internet connection at work and it took a while to download. I fear it would take a really long time on a dial-up link.

I had one problem with the installer program. Once all the packages were loaded, the post install part of the program seemed to just hang. I ran it again (which did not require re-downloading the files) and it also hung at the post install stage. I gave up and tried to run UQLX-win32. I got a few error messages about not finding the specific CYGWIN libraries and DLL files. To get it all fixed I had to add
"C:\cygwin\bin;C:\cygwin\usr\X11R6\bin"
to the default Windows path

Running UQLX-win32

I put UQLX-win32 in a single directory. In that directory is the .uqlxrc configuration file and a uqlx.bat file. I edited the .uqlxrc file as I needed it. Mostly I changed it to use a QXL.WIN file for win1_ instead of a subdirectory. The uqlx.bat file is a DOS batch file that first starts the X server

and then starts UQLX (the qm.exe executable).
On my system, UQLX-win32 comes up a little different. I first see a typical grey X-windows background with the X cursor. Then in the upper right of the screen the QL screen pops up. I have to move the cursor to the QL screen to type in it.

Floppy Disk Issues

I configured my .uqlxrc file to use A: for flp1_. For some reason it would not work. I contacted Pheobus Dokos and he make a few suggestions, but I still could not get it to work. The laptop I use, does not have room for both a floppy and CD drive, so the drives are hot pluggable and can be swapped out at any time. I don't know if this is a cause for the floppy issue, but I just could not get the system to read a QDOS formatted floppy.
The workaround is to use the 'dd' command that comes with Cygwin. With the floppy in the drive I use 'dd' to copy the contests of the floppy disk straight to a file:

```
dd conv=sync if=/dev/fd0 of=floppy1.bin
```

This commands reads a data stream from the floppy (/dev/fd0 in Cygwin terms) and sends it to a file. The 'dd' command will error out when it reaches the end of the floppy. Now I have a file that I can read with UQLX-win32 as a QDOS file system. Granted this is a little time consuming and I can't swap out floppies quickly, but at least I can now get QL executables to UQLX-win32.

Big Screens

I tried using the options to UQLX-win32 to run with larger screens, but every single one of them would give me a larger black screen, but I could not see the default QL screen, including the cursor.

BOOT

By default UQLX-win32 does not come with an mdv1 directory like regular UQLX does. I wondered if the code had been changed so that UQLX can now find a boot file on win1_. I created one, put it on win1_ and restarted UQLX, but no luck. I created an mdv1 directory, copied the boot program there and it was run as soon as I restarted UQLX. So in the manner of booting, UQLX-win32 works exactly the same an regular UQLX.

Running Applications with UQLX and UQLX-win32

I tried running a few applications to test the compatibility of UQLX and UQLX-win32 with QDOS. As mentioned above, the Pente game failed on UQLX but ran fine on UQLX-win32. I tried TURBO on both systems and it compiled an example program just fine.
I was going to use UQLX-win32 to tinker with writing TurboPTR programs. When I tried to run SETW, the program looked like it was trying to start, but nothing happened. When I tried to run TurboPTR compiled program, I got an "Can't Open Windows" error.. When I tried it under UQLX, SETW would run, but I think it was not running right and any TurboPTR compiled programs ran just fine.

Running SMSQ/E with UQLX and UQLX-win32

The final thing I wanted to test was if SMSQ/E could be run under either UQLX or UQLX-win32. I purchased a copy of SMSQ/E for the Gold Card. I put it on my laptop and LRESPRed it from UQLX-win32. The screen went a little funny, then I got the typical reboot multicolored screen and then I got the Minerva opening screen. When I hit the F1 key, the screen cleared, UQLX-win32 then looked like it went into Mode 8 and then nothing. No cursor, nothing. Ok, so that did not work.
For UQLX, I copied the file to the Linux box, ran UQLX and then LRESPRed the SMSQ/E binary. The end result, UQLX did nothing other than lock up.
So, it looks like neither UQLX or UQLX-win32 can run SMSQ/E.

Conclusion

The end result of all this testing is that UQLX and UQLX-win32 can give you a QL that is similar to a Trump Card-based QL. You don't have the level 2 drivers so that you can use "real" subdirectories, but you do have access to QXL.WIN files so that you have lots of storage space. Looking around for some of the utilities created to get around not having "real" subdirectories can probably improve the usablilty of the system.
One down side to UQLX-win32 is the need to install CYGWIN. This can take up a fair bit of hard drive space and some time to download the images. Once it's installed, it works fine.
The key advantage of UQLX and ULQX-win32 is that they are free. Both distributions come with Minerva 1.97 so you don't have to copy the ROM from your QL system. I did notice that UQLX-win32 did come with a version of TKII. I don't know if this is a freeware version of TKII or not, but it is there. The other necessary tool, qlxtool, is freely downloadable off the net.

Programming QPTR in SBASIC - Last part

W. Lenerz

Additional Commands

The QPTR extensions contain some additional S*Basic keywords, as follows:

I - Commands for the mouse and the hotkey system

Several keywords are concerned with the mouse and access to the hotkey system.

A - Accessing Hotkey System II

The hotkey system is closely linked to the Pointer Environment and two commands give you some access to it.

1) Filling the Hotkey buffer

The hotkey buffer (also called "stuffer buffer") is a small buffer that you can fill with strings which you can then get at by hitting the hotkeys ALT + SPACE (or ALT + SHIFT + SPACE) together. This, however, is only possible once the Hotkey job is running, which is achieved via the HOT_GO command of Hotkey System II (if you don't have the HOT_GO command, then you are still using Hotkey System I – an immediate upgrade is really necessary).

As soon as the hotkey is hit, the content of the stuffer buffer will be stuffed (hence the name) into the current keyboard queue (just as if you had used the old TK II Altkey system – please note that Hotkey System II will get rid of the Altkey used by TK II, else too many routines would compete for access to the Altkeys). The effect is that the string appears as if you had input it via the keyboard.

The stuffer buffer can also be filled by other programs: thus QPAC2's FILES menu puts the names of files selected into the stuffer buffer. So does QD with the names of the files saved/loaded. FiFi can also do this, and so can others (I would really like this to be a configurable feature of every program, though). Recent versions of SMSQ/E will also put a string currently being edited with the INPUT command, or by programs using the "edit line" trap, into the stuffer buffer whenever F10 is hit during editing.

With the HOT_STUFF command, you can explicitly put a string into the stuffer buffer. The syntax of this command is:

```
HOT_STUFF a$
```

a$ is the string to be put into the buffer. You can put several strings in there by passing them as parameters separated by commas:
`HOT_STUFF a$,b$,c$,d$....`
the string a$ will be put into the buffer first.

2) Picking a job

You now know that jobs (or their windows) are organized in a stack. The job the window of which is on top of the stack will have its window unlocked. With the PICK function, you can bring a job to the top, where its window will be visible and unlocked. This is like a repeated CTRL +C, but more targeted to a specific job. Instead of just cycling through all jobs as does CTRL + C, you can PICK any specific job you want.

The syntax of this function is:

```
result = PICK ([#channel,] JobID )   or:
result = PICK ([#channel,] key)
```

As usual, if you do not specify a channel number, channel #1 will be taken as default.

The job ID can be specified as "job number, job tag", which is what is returned by the TK II JOBS command. You may also use a single number:
`job_tag *65536 + job_number`

The "key" may be -1 or -2. If you use a key of -1, then the job at the bottommost place will be picked to the top. If you give -2 as key, then the same thing happens, but the window of that job will be marked as unlockable: its output will always be visible as soon as it changes.

B - Mouse commands

1 - Filling the mouse buffer

In a similar way that we have a Hotkey System II stuffer buffer, there is also a mouse buffer – but this is severely more limited. Indeed, the buffer

holds only two characters at the most. It can be filled with the MS_HOT command.

The content of the mouse buffer may be retrieved by clicking both mouse buttons at the same time – this buffer thus is only for those that do have a mouse...

The syntax of this command is:

```
MS_HOT [#channel,],a$
```

where a$ is a string of two characters at the most.

As usual, the channel number will default to #1 if you do not specify it.

If you pass an empty string then clicking both mouse buttons at the same time will no longer have any effect at all.

The interesting thing about the mouse buffer (and this is contrary to the stuffer buffer) is that the mouse buffer is polled before the Hotkey/ Altkey routines poll the keyboard. Practically, this means that you may use the mouse buffer character to set off a hotkey – when you click both mouse buttons, this behaves as if you had hit the corresponding hotkey. To achieve this, though, you must fill the mouse buffer with two characters, the first must correspond to the ALT key (i.e. CHR$(255)) and the second to the Hotkey you wish to activate.

2) Changing mouse speed and wake up

You may change the mouse speed and wake up time.

The mouse speed (or "acceleration") determines how far the mouse pointer moves on the screen whenever you move the mouse on your desk (or whatever). Grossly: if the speed is high, the pointer moves a lot with a feeble mouse movement. If the speed is low, the pointer moves less and you need to move the mouse a lot further to move the pointer on the screen. The speed also commands the gradual acceleration of the mouse pointer when the pointer is moved via the cursor keys rather than the mouse.

The mouse "wake up" is the mouse movement that is necessary to show the pointer on the screen when the pointer isn't already visible, for example if it is in a window that is waiting for keyboard input (blinking cursor). This can be easily seen in a Basic input window. The pointer ,normally isn't visible in that window, it becomes visible when you move the mouse. Try it, you will see what I mean.

The command for this is MS_SPD and its syntax is:

```
MS_SPD acceleration [,wake_up]
```

Both parameters range from 0 to 9 and the wake up parameter is an optional parameter.

You can also use the QPAC II "SYSDEF" menu and see how these two parameters change the behaviour of the mouse.

II - Commands for Blobs and Patterns.

Blobs and patterns were already defined in an earlier instalment of this series, please refer there if in doubt.

There are several commands which make the use and creation of blobs and patterns a bit easier:

A - Pattern creation

Here is a command that is useful to create a pattern of a bit more complicated design. Indeed, you may wish to design an image (for example with a painting program) and convert it into a pattern later on. This is pretty nifty as you don't have to care about how to make a pattern in the more complicated way. The command for this is MKPAT:

```
MKPAT address,buffer
```

-› * ***buffer*** is a buffer holding the painting, which was created, for example, with the PSAVE function (which was already covered in this series). The content of this buffer will be transformed into a pattern which will be put at address.

-› * ***address*** is the address in memory where the image converted into a pattern will lie. You must have reserved this address (for example with RESPR or ALCHP) and have enough space at the address for the resul-

ting pattern (including the header). This address can then be used whenever you need a pattern.

Thus note that you need to know the memory size for the pattern before you start this operation. You can get to know the necessary size by using the SPRSP function which we already have seen in an earlier instalment of this series - just use the x size of the buffer and half of the y size of the "buffer" - and then add 18 to take into acount the header.

The pattern (and the image in the buffer) must be at least 16 pixels wide (and the pattern will normally be cut to a length that is a multiple of 16 pixels).

B - Writing blobs and patterns

Once you have created a blob and a pattern you can "write" them out to the screen, i.e. have them appear anywhere you want. Please be reminded that a blob without a pattern, and a pattern without a blob are invisible.

1) WBLOB

This command writes a blob (Write BLOB) with its corresponding pattern to specific screen coordinates:

```
WBLOB [#channel,]x, y, blob, pattern
```

-› * obviously, **x** and **y** are the screen coordinates where the blob is to be written. 0,0 is the top left hand, and these coordinates are in pixels, relative to the window origin of the channel given as parameter.

-› * ***blob*** and ***pattern*** are, of course the pointers to the memory addresses where you can find the blob and pattern to be written out.

As usual, the channel parameter will default to #1 if it isn't specified. The blobs and patterns are written into the channel window at the specified coordinates. If the coordinates are outside the window, there is no error but the blobs and patterns will not be drawn. Pattern should be a multiple pf 16 pixels wide. Some (pretty old) versions of the Pointer Interface do NOT check whether the parameters are really blobs and patterns - if they aren't there is a good chance that the machine will crash. Hence - make sure!

2) LBLOB

The LBLOB (Line of BLOBs) command allows you to print one or several lines of blobs on the screen:

```
LBLOB [#channel,] xpos, ypos, blob, pattern
```

-› * ***xpos*** and ***ypos*** are the screen coordinates. You may combine them with the TO operator:
`xpos,ypos TO x1pos,y1pos (TO x2pos,y2pos etc)`
just like you would with the S*Basic LINE command.

-› * ***blob*** and ***pattern***, are the same pointers to blobs and patterns as described above.

3) SPRAY

This interesting little command is like WBLOB, but instead of writing an entire blob, it only writes out a random number of pixels of it. This is really only necessary in some kind of painting program, where, instead of drawing a continouous line, you would want to write out a more diffuse line. The "pencil" thus just leave a spray of pixels (hence the name) with a diffuse line.

```
SPRAY x, y, blob, pattern, pixels
```

-› * the first four parameters are like for WBLOB.

-› * ***Pixels***: This parameter gives the (approximate) number of quantity of pixels that will be drawn. However, even if you paint several times over the same place with the same pixel, you will not be sure that the entire blob will be drawn out (after all, you have WBLOB for that!)

This concludes this little series on QPTR. I hope you have enjoyed it more than I have....

Programming in Assembler - Part 11

Norman Dunbar

Linked Lists Demo Code

If you think back a couple of issues, you'll remember that I started a short series on Linked Lists. In that issue (Volume 9, Issue 1) I described singly linked lists and left you with a test harness for a demo routine promising that I would be providing the demo code to slot into that harness ready to test out our singly linked lists. Here is that very demo code.

The obligatory error report

As ever, I got some bits wrong in Volume 9 Issue 1, so here are the two corrections I need to make:

On page 22 at the top, I say '... then copy the value in the address that A1.L points to into A0 and ...' that 'A1.L' sould of course read 'A0.L' because that is the list pointer, not A1.

On page 23 of the same issue, there is a little bit of code missing from the code to find a node in a list. It currently reads like this:

```
FindNode    cmp.l #0,(a0)       ; Reached the end yet ?
            beq.s DelExit       ; Yes, node not found, exit with error

            move.l (a0),a3      ; Fetch the NEXT node address into A3.L
            jsr (a1)            ; And jump into the comparison routine
            beq.s FindExit      ; Looks like we found our node

FindNext    move.l (a0),a0      ; A0 now holds the NEXT node in the list
            bra.s  FindNode     ; Go around again

FindExit    tst.l d0            ; Set zero flag for success, unset for error
            rts
```

There is a need to clear out the D0 register when we have found the node we are looking for. If not, the Z flag is never set to indicate that the node was found. To remedy this, change the instruction:

```
            beq.s FindExit      ; Looks like we found our node
```

to the following instead:

```
            beq.s FindFound     ; Looks like we found our node
```

Then add the following one line of code immediately above the FindExit label:

```
FindFound   moveq #0,d0         ; Clear the error flag
```

Now when a node is found, we clear D0 and drop into the FindExit code to set the Z flag and exit. Good old QMON2 helped find that little problem.

Finally, there is a single line of code to remove from the code in the test harness itself. On page 28, there is a routine named 'Finished' which clears the screen at the end of the demo code. This one line should be removed, so change this code:

```
Finished    movea.l  con_id2(a4),a0      ; Title channel id
            bsr.s    cls                 ; Clear screen
```

to this code by deleting the second line shown above:

```
Finished    movea.l  con_id2(a4),a0      ; Title channel id
```

Now, when the demo is run, you will be able to see its output before the screen is cleared at the end of the program.

On With The Demo Code

The following code is all you need to insert into the test harness at the label 'Demo'. The code is a small example of creating and navigating a linked list. The demo starts by creating a list of 5 nodes, each holding one long word of data being simply the node number 0 to 4.

The list contents are then printed on the screen showing the node address, the next pointer and the data stored in that node. Once this is done, the node with data contents of 3 is located and deleted prior to the new list being printed out again.

Finally, each node in the list is deleted.

To add this code into the test harness, copy the test harness to SingleList_asm, open that file in your favourite editor and locate the stub routine for the Demo code and replace all of it with the following.

The first part of the code simply controls the whole demo by calling various sub-routines to do the hard work, display messages etc on screen.

```
* ==========================================================================
* The DEMO code starts here.
*
* This demo does the following:
*
* Creates a number of nodes and stores a LONG value in each one.
* Lists each node address, it's NEXT pointer and data value on screen.
* Deletes a node.
* Lists each node address, it's NEXT pointer and data value on screen.
* Finds a node based on its data value and displays its details on screen.
* Deletes all the nodes from the list.
* ==========================================================================
Demo        bsr     BuildList           ; Build a linked list
            bsr     Before              ; Display 'BEFORE :'
            bsr     ShowList            ; Display list details
            bsr     FindNode            ; Locate a single node
            bne.s   DemoAfter           ; Failed to find node with data = 3
            bsr     DeleteNode          ; Delete a single node

DemoAfter   bsr     After               ; Display 'AFTER : '
            bsr     ShowList            ; Show details again
            bsr     KillList            ; Delete entire list
            rts                         ; Done
```

Following on from the main control section of the demo, we have our much beloved root node which must be initialised to zero as outlined in the original article. This is the pointer we will be loading into A0 quite often in the demo and it holds the address of the first node in the list. At present, there is no list, so the contents are set to zero to indicate the very end of the list.

```
* ---------------------------------------------------------------------
* A location to hold a single long word pointing to the first 'real'
* node in our linked list. This must be initialized to zero.
* ---------------------------------------------------------------------
RootNode    dc.l    0                   ; This is our root node.
```

The first of our sub-routines follows on. This part builds a list of 5 nodes in the most simple manner possible - it runs a loop which calls the sub-routine to create a single node and link it into the list. If you want a bigger list, change the counter loaded into D7 to one less than the number of node you want. Don't forget to adjust the height of your window as well if you want to see all the results on screen at the same time.

```
* ---------------------------------------------------------------------
* Build a list of 5 nodes each holding a long word of data.
* ---------------------------------------------------------------------
BuildList   lea     RootNode,a0         ; Pointer to root node address
            moveq   #4,d7               ; How many nodes in D7 = 5 (DBRA remember ?)
```

```
            moveq   #8,d1               ; Each node is 8 bytes long

BuildLoop   bsr.s   BuildNode           ; Create a new node, address is in A1.L
            bne     all_done            ; Just die on errors
            move.l  d7,4(a1)            ; Store data value - just our counter
            bsr.s   AddNode             ; Add to list
            dbra    d7,BuildLoop        ; Do the rest
            rts                         ; Done
```

Here's the first of the real list routines. We add a new node into the list in the manner outlined in the article. We reject attempts to add the root node into the list - but without flagging any errors - and, as explained, we don't attempt to check if the new node already exists in the list because we are creating the node on the heap, so the chances of the new node being present already are pretty slim to say the least.

```
* ------------------------------------------------------------------------
* AddNode - Adds a new node to a list. See text for details.
* Preserves all regsiters.
* No errors returned.
* ------------------------------------------------------------------------
AddNode     cmpa.l  a0,a1               ; Don't allow the root node to be added again
            beq.s   AddExit             ; Bale out quietly if attempted
            move.l (a0),(a1)            ; Save current first node in new node's NEXT area
            move.l  a1,(a0)             ; Store address of new node in FIRST storage area
AddExit     rts
```

A new node is built by allocating some space on the common heap but we must preserve the working registers, the following code does this for us.

```
* ------------------------------------------------------------------------
* Allocate a single new node
* On entry, D1.L is amount of memory required.
* On exit, A1 holds the address of the new node, with D0 holding errors.
* All registers preserved - unless otherwise stated.
* ------------------------------------------------------------------------
BuildNode   movem.l d1-d3/a0/a2-a3,-(a7) ; Save working registers
            moveq   #MT_ALCHP,d0        ; Set the trap
            moveq   #me,d2              ; I want it for me
            trap    #1                  ; Do it
            move.l  a0,a1               ; Get the node address where we need it
            movem.l (a7)+,d1-d3/a0/a2-a3 ; Restore working registers
            tst.l   d0                  ; Set flags
            rts                         ; Exit
```

The following sub-routine is called once to display the linked list before we do the deletions and again after we have deleting a node. The code simply walks through the entire list and prints out the node address, the next pointer and the data value by calling other sub-routines.

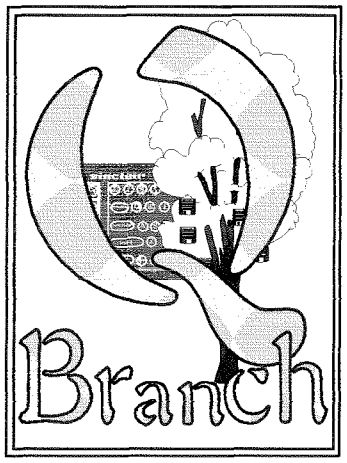
```
* ------------------------------------------------------------------------
* Walk through a linked list displaying the details of each node as we
* go.
* On entry, A0 = root node of the list.
* ------------------------------------------------------------------------
ShowList    lea     RootNode,a0         ; Root node address

ShowLoop    move.l (a0),a0              ; Get address of the next node
            cmpa.l  #0,a0               ; Done ?
            beq.s   ShowExit            ; Yes
            move.l  a0,-(a7)            ; We must preserve A0 - it's our node pointer
            bsr.s   ShowNode            ; Display that node's details
            move.l (a7)+,a0             ; Restore the node pointer again
            bra.s   ShowLoop            ; Do the rest of the list

ShowExit    rts                         ; Done
```

This next short routine is called with the address of a node in A0.L and prints the details of that node to the screen.

Q Branch



Text 87

£ 79.00

Typset 94 £ 29.00
Fountext 94 £ 39.00
2488 drivers £ 29.00
Epson ESC/P2 drivers £ 26.00

Text 87 High Colour Patch Available Now

£9.00

PROGRAMMING

QD 2003	£ 49.00
QD + QBasic	£ 63.00
QD + Qliberator + QBasic	£ 104.00
Qliberator	£ 50.00
Master Spy v 3.3	£ 30.00
QPTR	£ 32.00
Easyptr pt 1 & 2 (together)	£ 30.00
Easyptr pt 3 (C library)	£ 14.00
QMake	£ 18.00
QMon / JMon	£ 22.00
Basic Linker	£ 22.00
DISA 3	£ 34.00
QMenu	£ 16.00

Yet Another High Colour Update

We now have a full high colour version of QPAC 1 available. If you need to update your copy then send it to with the appropriate copying fee. I would like to point out here that many people send me disks for updating in just a plain envelope and this can lead to damage to the disk. Please send disks well packed in a padded envelope and preferably surrounded by cardboard. The english phone company BT have recently send out advertising material in the form of fridge magnets. This has not helped the passage of magnetic media in the post.

As we prepare this advert I have just heard that the entire ProWesS suite may be becoming freeware. More details in our next adverts.

Available Now!

High Colour Drivers for Gold /Super Gold Card SMSQ/E

(Aurora only - Send old master disk as proof of purchase)

£25.00

These programs have been updated to High Colour versions:-

QMake
WinEd
FiFi II
QPAC 1
QPAC 2
QSUP

<<<<< SuQcess >>>>

Please send old master disk and £ 1.60 to cover postage and copying fee.

QL Keyboard Membranes

Last Few!

£ 17.50 ea + P&P

DATAdesign

High Colour Update **£8.00**

UTILITIES

FiFi 2	£ 21.00
QSup	£ 30.00
QSpread 2003	£ 51.00
Cueshell 2	£ 15.00
Qload / Qref	£ 15.00
Disk Mate 5	£ 16.50
QPAC 1	£ 22.00
QPAC 2	£ 42.00
QTYP 2	£ 31.00
QLQ	£ 30.00

These Programs are paid Upgrades:

QSpread 2001 to 2003	£ 10.50
QD 98 to 2003	£ 10.50
QPC2	-
update from v3.xx	Free
update from v2	£ 13.90
update from v1	£ 34.00

Return Master Disk With Order

```
* ------------------------------------------------------------------------
* Display the details of a single node in the linked list.
* On entry A0 = the node address.
* ------------------------------------------------------------------------
ShowNode    move.l a0,a5                ; The node address
            move.l con_id2(a4),a0       ; The channel address
            move.l a5,d4                ; The node address
            bsr.s  ShowAddr             ; Print node address
            move.l (a5),d4              ; The NEXT pointer
            bsr.s  ShowNext             ; Print NEXT pointer
            move.l 4(a5),d4             ; The node data
            bsr    ShowData             ; Print the data
            rts
```

Obviously, just displaying the list contents isn't very user firiendly, so the next couple of routines display a title which informs the user if the list being displayed is 'before' or 'after' the deletion of a node.

```
* ------------------------------------------------------------------------
* Display 'BEFORE :' in the output channel.
* ------------------------------------------------------------------------
Before      lea     BeforeAddr,a1       ; The prompt
            movea.l con_id2(a4),a0      ; Needs channel id
            bsr     Prompt              ; Print it
            rts

BeforeAddr  dc.w    B4End-BeforeAddr-2
            dc.b    'BEFORE :',linefeed
B4End       equ     *

* ------------------------------------------------------------------------
* Display 'AFTER :' in the output channel
* ------------------------------------------------------------------------
After       lea     AfterAddr,a1        ; The prompt
            movea.l con_id2(a4),a0      ; Needs channel id
            bsr     Prompt              ; Print it
            rts

AfterAddr   dc.w    AftEnd-AfterAddr-2
            dc.b    linefeed,linefeed,'AFTER :',linefeed
AftEnd      equ     *
```

There now follows one of three separate but short routines to display on screen, the various parts of a list node. This one simply displays the node's address in memory. Following after this routine is a number of small sub-routines which assist in the displaying of node data by converting the contents of D4 to hex and printing it to the screen.

```
* ------------------------------------------------------------------------
* Display the node's actual address in memory.
* On entry D4 = the node address.
* ------------------------------------------------------------------------
ShowAddr    lea     MsgAddr,a1          ; Our prompt

ShowPrompt  bsr     Prompt              ; Print it
            bsr.s   D4ToHex             ; Convert D4.L to hex
            bsr.s   PrintHex            ; Print it and a linefeed
            rts

MsgAddr     dc.w    AddrEnd-MsgAddr-2
            dc.b    linefeed,'Node address : '
AddrEnd     equ     *

* ------------------------------------------------------------------------
* Print the contents of the buffer to screen.
* ------------------------------------------------------------------------
PrintHex    lea     Buffer,a1           ; What to print
            move.l  con_id2(a4),a0      ; Channel to print to
            bsr     Prompt              ; Do it
            rts
```

```
* ------------------------------------------------------------------------
* Convert the long word in D4 to hex ready for printing
* ------------------------------------------------------------------------
D4ToHex     lea     buffer+2,a1         ; Buffer address
            bsr     hex_l               ; Do all 4 bytes = 8 characters
            lea     buffer,a1           ; Buffer again
            move.w  #8,(a1)             ; Store text length
            rts
```

The second and third routines to diplay the details of a node now follow. Starting with the code to show the node's NEXT pointer address closely followed by the code to print the actual data stored in the node.

```
* ------------------------------------------------------------------------
* Display the node's NEXT address in memory.
* On entry D4 = the node's NEXT pointer.
* ------------------------------------------------------------------------
ShowNext    lea     MsgNext,a1          ; Our prompt
            bra.s   ShowPrompt          ; Print it

MsgNext     dc.w    NextEnd-MsgNext-2
            dc.b    '  NEXT pointer : '
NextEnd     equ     *

* ------------------------------------------------------------------------
* Display the node's actual data content.
* On entry D4 = the data.
* ------------------------------------------------------------------------
ShowData    lea     MsgData,a1          ; Our prompt
            bra.s   ShowPrompt          ; Print it

MsgData     dc.w    DataEnd-MsgData-2
            dc.b    '  Data value : '
DataEnd     equ     *
```

Next we have the code to locate a single node in the linked list based upon the data part of the node. This part is simply the setup routine for the following code at FindANode which does the actual scanning of the node and calling the compare routine as described in the original article.

```
* ------------------------------------------------------------------------
* Locate a node in the list based on it's data value.
* On exit, A1 is the required node's address plus Z set - if found.
*          A1 is undefined plus Z clear - if not found.
* ------------------------------------------------------------------------
FindNode    lea     RootNode,a0         ; Pointer to root node in list
            lea     Compare,a1          ; Address of node comparison routine
            moveq   #3,d1               ; The data value we are looking for
            bsr.s   FindANode           ; Go find it, or not
            rts

* ------------------------------------------------------------------------
* This routine expects the following input registers so that it can scan
* a linked list for the required data value and return the address of the
* node holding that data value with the Z flag set if found, or the Z flag
* cleared for not found.
*
* A0.L = Rootnode of the list.
* A1.L = Address of Compare routine.
* D1.L = Value to look for in list.
* ------------------------------------------------------------------------
FindANode   moveq   #oops,d0            ; Assume not found (yet)

FindLoop    cmpa.l  #0,a0               ; Reached the end yet ?
            beq.s   FindExit            ; Yes, node not found, exit with error

            move.l  (a0),a3             ; Fetch the NEXT node address into A3.L
            jsr     (a1)                ; And jump into the comparison routine
            beq.s   FindFound           ; Looks like we found our node
```

```
FindNext    move.l  (a0),a0             ; A0 now holds the NEXT node in the list
            bra.s   FindLoop            ; Go around again

FindFound   movea.l a3,a1               ; This is the required node
            moveq   #0,d0               ; Clear error flag

FindExit    tst.l   d0                  ; Set zero flag for success, unset for error
            rts

* ------------------------------------------------------------------------
* This is the simple compare routine for our FindNode code. On entry, we
* have the following registers set :
*
* D1.L = The value we want to find in a node in the list.
* A3.L = The address of a node which we are checking for the data value.
*
* We must preserve A0, A1 and D1.
*------------------------------------------------------------------------
NData       equ   4                     ; Offset into a node to the data part.
Compare     cmp.l NData(a3),d1          ; Is the data in the node = the value we want?
            rts                         ; Exit with Z set if so, unset otherwise.
```

This next routine is not really required on QDOSMSQ as a terminating job always has any allocated heap areas returned to the system by the job scheduler routines. Because I'm a lazy typist and in order that I reduce the large amounts of code in the magazine, I'm not writing any code here!

If you wish to carry out the list tidying explicitly for yourself as an exercise, feel free to do so. As a suggestion, start a loop which keep going around the list fetching the NEXT node pointer and deleting that from the list using the routines in this code. Once the node has been unlinked from the list, you may deallocate it's heap area - but don't forget to preserve those registers.

```
* ------------------------------------------------------------------------
* QDOSMSQ tidies up rather nicely for us on exit - so we don't have to !!
* ------------------------------------------------------------------------
KillList    rts
```

The following code sets up the demo to delete the node that was just 'found' by searching for the node holding data 3. This code is called with the address of the '3' node in A1.L and it simply sets up the following routine which actually scans the list looking to make sure that the node we are deleting exists in the list.

```
* ------------------------------------------------------------------------
* A demo routine to delete the node whose address is passed in A1.L. Sets
* Z if found &deleted, clears it otherwise.
* ------------------------------------------------------------------------
DeleteNode  lea     rootnode,a0         ; Address of the root node
            bsr.s   DelANode
            rts
```

This is the node deletion code itself. As described in the article, we must not delete the root node itself - as this isn't allocated on the heap. We must also check that the node is in the list by scanning from start to finish looking for the node in the list which has a NEXT pointer holding the address of the node we want to delete.

We remove a node from the list by copying the soon to be deleted node's NEXT pointer into the NEXT pointer of the node before it, thus bypassing the node we want to delete.

BEWARE this code only deletes a node from the linked list. It does not deallocate the memory on the common heap that was allocated to create the node. QDOSMSQ will do this at the end of the demo, but in real life, you would need to cary out this task yourself - expecially as you may not want a number of deleted heap areas hanging around in memory fragmenting your heap.

```
* ------------------------------------------------------------------------
* Routine to delete a node whose address is passed in A1.L from the list
* whose address is passed in A0.L. On exit, Z flag will be set if deleted
```

```
* or cleared if not.
* ------------------------------------------------------------------------
DelANode    moveq   #oops,d0            ; Assume it's going to fail
            cmpa.l  a1,a0               ; Trying to delete the rootnode ?
            beq.s   DelExit             ; Exit if so.

DelLoop     cmpi.l  #0,(a0)             ; Finished yet ?
            beq.s   DelExit             ; Exit not found
            cmpa.l  (a0),a1             ; Found the previous node to the one
*                                       ; we want to delete ?
            bne.s   DelNext             ; Not yet, try again

DelFound    move.l  (a1),(a0)           ; Delete the node by setting the NEXT
*                                       ; pointer to the node BEFORE the one to
*                                       ; be deleted to the NEXT pointer of the
*                                       ; node that is being deleted.
            moveq   #0,d0               ; Indicate found and deleted
            bra.s   DelExit             ; Set Z flag on the way out

DelNext     move.l  (a0),a0             ; Get the next node in the list
            bra.s   DelLoop             ; And try again

DelExit     tst.l   d0                  ; Set or clear Z flag
            rts

* ========================================================================
* The DEMO code ends here.
* ========================================================================
```

And that is all there is to it. The SingleList demo code should be assembled and run in the normal fashion. You'll be able to see that there are indeed 5 nodes in the list (in the BEFORE section at the top of the screen) then under that, the AFTER section shows a missing node with data content 3 - we have deleted it from the list.

Next time, we'll have another 'drop in' demo code which shows the use and abuse of doubly linked lists as discussed in the previous issue Volume 9 Issue 2. Until then, have fun.

Counting QLs

Simon N. Goodwin

While some of us still have the old black boxes in our possession, two decades after mass production of the QL started, it seems worth trying to answer the old question of how many QLs were made.

Computer Trade Weekly reckoned QL sales were between 100,000 and 200,000, in a post-Sinclair article published on 5th October 1987. It's unlikely that any more machines were made after this date, but sales continued for years after, and this figure may only count those made in the UK, for sale in Europe; the QL was also manufactured by Samsung in Korea, mainly for export to the USA.

The original production schedule agreed with Thorn EMI Datatech called for production of 20,000 machines a month in mid 1984, according to a report in the Times when the QL was launched (13th January 1984), with potential to ramp up production to 100,000 a month if demand was that great - a figure presumably set from the peak demand for Sinclair's previous micro, the ZX Spectrum.

According to Popular Computing Weekly (in an article published on 7th Febrary 1985) Sinclair sold 44,297 QLs in the first year. This is backed up by a comment in Rodney Dale's book The Sinclair Story which says fewer than 60,000 sold in 1984. More than 13,000 of those were ordered in the first part of the year, before the first complete machines were delivered - a few hundred were shipped at the end of April, with part of the operating system and SuperBASIC in the infamous kludge hanging out of the back of the machine, but production only got into four figures at the end of May when Thorn EMI shipped machines with the AH ROM, actually three individually programmed 16K EPROMs with two of those piggy-backed to share one of the sockets inside the machine.

ROM wrangling

The sockets were wired for two 16K mask-programmed ROMs, back when Sinclair planned to put GST's windowing QL operating system in 32K of internal memory and load SuperBASIC from cartridge when required. This plan changed when Psion were having trouble getting their bundled packages to fit into RAM and run reliably from tape; GST's operating system missed the boat, though it's screen output routines were used in Qdos, and it was eventually available as an obscure option for QL completists. Sinclair decided to put Jan Jones's SuperBASIC in ROM along with Tony Tebby's 'skunkworks' fallback project Qdos, but needed more space than the original - hence the 16K 'kludge' that occupied the ROM socket of the first few QLs shipped with very early EPROMs before the QL was adapted to put one 32K and one 16K chip in its internal sockets, rather than the originally-planned pair of 16K parts plus BASIC on cartridge.

Incidentally the first version of Tebby's QL Toolkit was written to fill the remainder of the half-used extra ROM chip - "AH" and "JM" versions of Qdos weighed in at a little over 40K - but later versions of the ROM soaked up the free space with bug-fixes, support for international variations (in the MG series) and a shaky implementation of features like WHEN event handling, that were designed but not implemented for earlier ROMs.

Care Electronics persuaded Tebby to pad the toolkit to fill a 16K external cartridge, and the dongle was back, though in a more elegant case, and SuperToolkit 2 was born - as he said at the time 'It's like breakfast: some people do without, but if you don't have it, you miss it.'

Once the ROM issues were sorted out, and the first expensive EPROMs were replaced with mask-programmed chips, in "AH", "JM" and the final UK "JS" versions, production was stepped up, and 8,200 machines with the proper ROM chips, one each of 16K and 32K, shipped in the last month of 1984.

Computer Dating

You can work out the month of manufacture of a QL, and the approximate number of machines made up to that point, from the serial number embossed on the base of the QL.

My first machine, one of the batch shipped at the end of May 1984, was a dud and didn't work for more than about 45 minutes, probably due to faults in the ZX-8301 ULA. Within a month it was replaced by one marked D06-006556 - the numbers indicating manufacture in July 1984, after (in theory) 6555 other QLs. In practice, according to Tony Tebby, machines with serial numbers less than 4000, all with the D00 to D04 prefix, were prototypes with various faults that made them unsaleable.

Another reason the serial number does not relate directly to the number of machines sold is that some failed testing - though in most instances the expensive cases were re-used - and thousands were returned as faulty, though a lot of those made it back into circulation when Sinclair assets were sold off in 1987.

Sinclair's general policy was to ship one of the machines from the current production line in return for a faulty computer, and put the broken one into storage for later investigation - which typically didn't happen till after Sinclair ran out of cash. At the time of Amstrad's takeover of Sinclair there were said to be a million machines in 'stock', most of those ZX Spectrums, valued on Sinclair's balance sheet at about £30, which was marginally more than Sinclair paid Timex in Dundee to make a Spectrum; a lot of those were customer returns, many from catalogue purchasers who never mastered cassette loading but in essentially good condition. Some of those were QLs, though only a minority.

The return rate for Spectrums rose as high as 25 per cent at times - typical for early 1980s micros, brought on by loading problems and customer ignorance as well as poor quality control - but that for QLs was around 3.5 per cent, putting the number of dud machines in the low thousands.

Job-lot box-shifters PST paid Amstrad £3M for 160,000 'Spectrums and QLs' in 1986, according to The Guardian newspaper (Computer Guardian 20th February 1986) - less than £20 a lump. Traders who bought those were upset to find out that some of them were broken and some were just

empty boxes - stock control was not a strong point for Sinclair or Amstrad. After the Amstrad takeover the upward procession of Sinclair serial numbers was disrupted, and even went backwards for a while, as warehouses were emptied in the reverse order that they'd been filled, with the oldest machines with the lowest serial numbers emerging last.

UK QL production seems to have run at about half the original projection in 1985, with 10,000 machines emerging a month till Sinclair ran out of cash. A D14 (early 1985) machine had a serial number of 95,934. Two months later one with a serial number of D16-113829 was made - I got it at a radio rally when machines used by a university Chemical Engineering department were sold off.

It would be interesting to collate the serial numbers of as many QLs as readers can be bothered to dig out and pass on details about. If you send QL Today the numbers from the bottom of your black boxes, we should be able to work out the number of machines made each month and the total number of QLs, at least more accurately than the CTW estimate.

Korean Kopies

Then there are the QLs made by Samsung in South Korea, rather than by Thorn EMI in the UK. Thousands of these were produced, initially for the North American market. The design was tweaked to suit, and made more reliable in the course of the vast Korean firm's production engineering work.

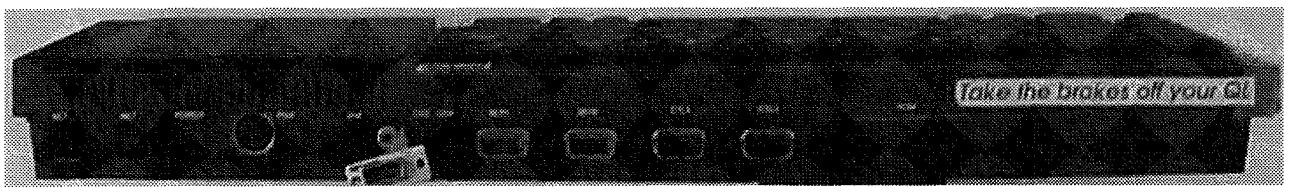
The visible differences are mainly on the back panel - the obscure but presumably very cheap six pin SER and CTRL sockets were replaced with better-made 9 pin D-type male and female sockets and plugs, for joysticks and serial peripherals respectively.

An oscillator was added to the ZX-8301 video circuit; along with a change of TV modulator, this enabled a VHF video output switchable between two channels. The inside of the case was screened to conform to the US FCC emissions regulations, much reducing digital interference with AM radio and TV sound common in the United States.

I was given a Samsung QL when the post-Sinclair US distributors of the QL, A-Plus Computer response, commisioned a de-Lenlocked version of Supercharge to bundle with what they promoted as a '32 bit Supercomputer' (for just $188) in the glossy code hacker magazine Dr Dobbs Journal. I never got paid for this work (I wonder if anyone ever got the special Supercharge, which had a few K of extra space for programs, valuable on a 128K system, thanks to the removal of the Lenslok code and data?) but was happy to add another rare computer to my collection.

I got that to work with a UK telly, though only in shades of grey, using a resistor and capacitor chosen by experiment to match the Samsung QL's NTSC video output to my PAL equipment (via my ZX Spectrum's modulator, the input to which I'd diverted out of the box to derive Composite PAL from that).

It was easy to tie the compiler into the US product as it had a JSU ROM with reduced-size font (8x6 rather than 10x6 pixels, so the whole TV screen fitted into the 192 visible lines on an NTSC TV, and lines of text in monitor mode were more cramped and widely spaced between the lines) and an extra control bit in the ZX-8301: bit 6 of MC-STAT, next to bit 7 which Minerva uses to select its second screen, is unused on Sinclair's European hardware but selects NTSC format video if set on a Samsung QL). This bit selects 192 line TV or 256 line monitor output, triggered from the initial F1/F2 selection.



The back panel of a Samsung QL, with 9 pin D type rather than 6 pin BT sockets. The added socket close to the VHF video output and channel switch is for connections to Tony Firshman's internal I2C interface. The sticker at the microdrive end is another add-on specific to Simon's machine - the slogan from the 1988 Speedscreen marketing campaign.

My Samsung QL is serial number S13-003505. It's the only one I've seen in the UK. But I know thousands were made - A-Plus bought at least 1500, probably more, when Sinclair's US office closed - but I'm not sure how many or for how long production continued. Some of the Samsung production run was later sold in Continental Europe, with the improved connectors, but I don't know if the TV output was switched back to PAL. I expect so, and would be interested to hear details from owners either side of the Atlantic, via QL Today.

My guess is that the Samsung agreed to build an initial batch of 10,000 machines, and when US sales were slow some of those were adapted for sale elsewhere. But if we collect some serial numbers we may yet be able to work out how many QLs were made, and where they ended up.

Columns in Quill

Dilwyn Jones

This program is a fairly simple basic listing for you to type in, which will help to create columns of text in Quill, or can arrange a plain text file into columns for printing. Figure 1 shows the results of creating a text file with three columns of text imported into Quill.

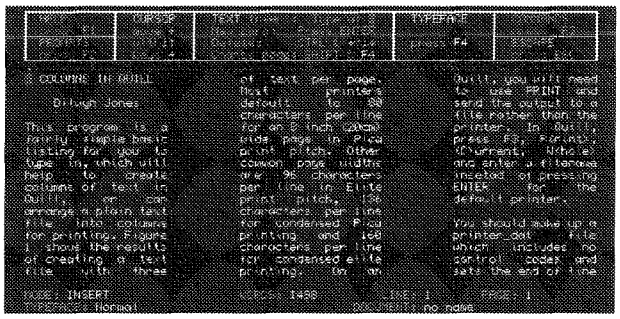


Figure 1 - Quill, displaying a text in 3 columns

The input file must be a plain text file (i.e. no bold, underline etc control codes, although you can manually apply these later if you import the columns of text into Quill for printing), with lines already prepared in the correct column width. You can use left justify, centre or right justify. The program outputs these lines arranged into columns. The output file is also a plain text file which can be imported into Quill (F3, O(ther), F(iles), I(mport)) for touching up if required, or for printing.

You will need to apply a little thought beforehand as to the format you want:

How many columns will fit onto a page What width (i.e. how many characters per line) the columns should be

The most common requirement will be two or three columns of text per page. Most printers default to 80 characters per line for an 8 inch (20cm) wide page in Pica print pitch. Other common page widths are 96 characters per line in Elite print pitch, 136 characters per line for condensed Pica printing and 160 characters per line for condensed elite printing. On an Epson compatible printer, here are the control codes to send to the printer before using this program, assuming the printer is on channel #3:

Pica, 80 characters per line (10 characters per inch or per 2.54 cm)

```
PRINT #3,CHR$(27);'P';
```

Elite, 96 characters per line (12 characters per inch or per 2.54 cm)

```
PRINT #3,CHR$(27);'M';
```

To set **Condensed** printing, just add: `PRINT #3,CHR$(17);`

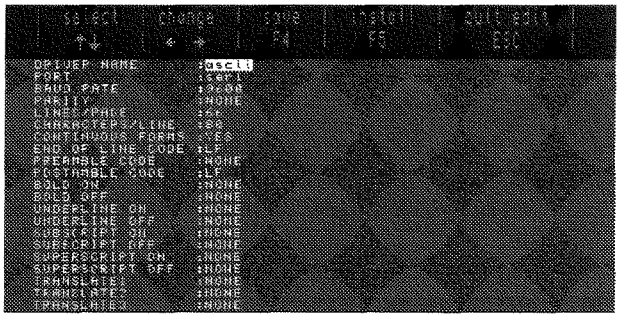
To cancel Elite printing, send the codes for Pica printing above.

To cancel condensed printing:

```
PRINT #3,CHR$(18);
```

To generate the text in the first place, load the text file and reformat into the line width you require for your columns. Commonly, this will be 20 or 30 characters wide. Save a copy of this text from your editor in this line width. If using Quill, you will need to use PRINT and send the output to a file rather than the printer. In Quill, press F3, P(rint), (C)urrent, W(hole) and enter a filename insetad of pressing ENTER for the default printer.

Figure 2 - Generating a suitable printer_dat for Quill with Install_bas



You should make up a printer_dat file which includes no control codes and sets the end of line character to just a linefeed, rather than Quill's default of Carriage Return + Linefeed. See figure 2 for a typical driver created with install_bas.

As Quill has a habit of generating files which do not end with a linefeed right at the end, you will need to press ENTER at the end of the file in Quill to force it to do so, otherwise it willo cause an 'EOF' error when an INPUT statement tries to read the last line in the file.
It is important that no control codes are included in the file which this program is to process. They will upset the column formatting.
Although this program does little more than rearrange text of the correct line width into columns, and is fairly primitive and not error protected in its current form, it should provide a skeleton program for you to develop according to your requirements.

Figure 3 is the listing of the BASIC program. It should work in either SuperBASIC or SBASIC.

Figure 4 shows it running. You can see that it asks for the following entries:

1. Number of characters per line on the printer. Normally, this will be 80, 96, 136 or 160 for an 8 inch (20cm) wide page printer, whatever you currently have your printer set to.
2. Number of lines per page. This will usually be 60, 66 or 70, depending on the size of paper you use and what top and bottom margins have been set for the printer.
3. Column width. This is the number of characters across each column - 20 or 30 are fairly typical values, although as long as the column width multiplied by the number of columns will fit on the page it doesn't really matter, use

Figure 3 - The program listing

```
100 REMark multi column text printing
110 REMark prepare plain text files in required column width
120 :
130 WINDOW 512,202,0,0 : WINDOW #0,512,42,0,202
140 BORDER 1,255 : BORDER #0,1,255
150 CLS : CLS #0
160 PRINT 'MULTI COLUMN TEXT'
170 PRINT
180 INPUT 'Number of characters per line    › ';cpl
190 INPUT 'Number of lines per page         › ';lpp
200 INPUT 'Column width (characters)        › ';cw
210 INPUT 'Number of columns                › ';columns
220 PRINT 'Left Margin: 0=none, 1=50% of'
230 INPUT '  the gap between columns        › ';lm
240 IF lm = 0 THEN
250   REMark no left margin
260   gap = (cpl-(cw*columns)) DIV (columns-1)
270 ELSE
280   gap = (cpl-(cw*columns)) DIV columns
290   lm = gap DIV 2
300 END IF
310 INPUT 'Filename of text file            › ';textfile$
320 INPUT 'Print to (e.g. SER1 or PAR)      › ';printer$
330 INPUT 'Preview Y/N ';prev$
340 IF prev$ == 'y' THEN Print_Preview
350 CLS : CLS #0
360 :
370 IF printer$ ‹› '' THEN OPEN_NEW #4,printer$
380 :
390 REMark set printer text pitch here if required, e.g. for
    Epsons:
400 REMark for 80cpl Pica : PRINT#3,CHR$(27);'P';
410 REMark for 96cpl Elite: PRINT #3,CHR$(27);'M';
420 REMark for Condensed, add PRINT #3,CHR$(17);
430 REMark Condensed Pica is 136 characters per line (cpl)
440 REMark Condensed Elite is 160 characters per line (cpl)
450 :
460 REMark get the text into page array ready to print
470 OPEN_IN #3,textfile$
480 page_no = 1 : CLS
490 REPeat page
500   CLS : PRINT 'PRINTING PAGE ';page_no
510   REMark last dimension of col$() must be an even value
520   DIM col$(lpp-1,columns-1,cw+(cw MOD 2)) : REMark round
      up to even
530   FOR column = 0 TO columns-1
540     FOR lne = 0 TO lpp-1
550       IF EOF(#3) THEN
560         txt$ = FILL$(' ',cw-LEN(txt$))
570       ELSE
580         INPUT #3,txt$
590         IF LEN(txt$) ‹ cw THEN txt$ = txt$&FILL$(' ',
            cw-LEN(txt$))
600       END IF
610       col$(lne,column) = txt$
620     END FOR lne
630   END FOR column
640   :
650   REMark preview the output
660   FOR lne = 0 TO lpp-1
670     AT 1,0 : PRINT'LINE NUMBER ';lne+1
680     PRINT #4,FILL$(' ',lm);
690     FOR column = 0 TO columns-1
700       PRINT #4,col$(lne,column);
710       IF column ‹ columns-1 THEN PRINT #4,FILL$(' ',gap);
720     END FOR column
730     IF printer$ ‹› '' THEN PRINT #4,
740   END FOR lne
```

```
750   :
760   PRINT #4,CHR$(12); : REMark form feed at end of page
770   IF EOF(#3) : EXIT page : REMark all done
780   page_no = page_no + 1
790 END REPeat page
800 REMark cancel text pitch etc here if required
810 REMark Restore Pica print pitch PRINT #3,CHR$(27);'P';
820 REMark Cancel condensed print   PRINT #3,CHR$(18);
830 CLOSE #3
840 IF printer$ <> '' THEN CLOSE #4
850 STOP
860 :
870 DEFine PROCedure Print_Preview
880   LOCal page_no,previewing
890   CLS
900   OPEN_IN #3,textfile$
910   page_no = 1
920   REPeat previewing
930     CLS: BLOCK cpl+8,lpp+8,0,10,7 : REMark page background
940     AT 0,0 : PRINT 'PAGE ';page_no
950     DIM col$(lpp-1,columns-1,cw+(cw MOD 2)) : REMark round
        up to even
960     FOR column = 0 TO columns-1
970       FOR lne = 0 TO lpp-1
980         IF EOF(#3) THEN
990           txt$ = FILL$(' ',cw)
1000          ELSE
1010            INPUT #3,txt$
1020            IF LEN(txt$) < cw THEN txt$ = txt$&FILL$(' ',
                cw-LEN(txt$))
1030          END IF
1040          col$(lne,column) = txt$
1050          x = 4+(lm+(column*(cw+gap)))
1060          y = 10+4+lne
1070          FOR c = 1 TO LEN(col$(lne,column))
1080            IF col$(lne,column,c) <> ' ' THEN
1090              BLOCK 1,1,x+c-1,y,0
1100            END IF
1110          END FOR c
1120        END FOR lne
1130      END FOR column
1140      IF EOF(#3) THEN EXIT previewing : REMark reached end
          of text
1150      CLS #0
1160      INPUT #0,'Next page Y/N ';yn$
1170      IF yn$ == 'n' THEN EXIT previewing
1180    END REPeat previewing
1190    CLOSE #3
1200    INPUT #0,'Finished previewing, press ENTER. ';yn$
1210 END DEFine
```

whatever width you require.

4. Number of columns - usually 2 or 3, although as long as the column width times the number of columns will fit on the page, it's just a matter of appearance. Obviously, silly settings like 40 columns each 2 characters wide will not be much use!
5. Left margin. This can either be set to 0, or the program will create a default value of half the gap between other columns.
6. Filename of the text file to be laid out in columns. This is the file you saved or printed in the line width you required - if you wanted columns 20 characters wide, you'd have saved the text in a file with each line no wider than 20 characters.
7. Where to print or save to. If you enter SER1, SER2, or PAR you can print direct to the printer. If you prefer to save the file in its new format in columns, enter a filename instead.
8. Do you want an on-screen preview? Enter a y for yes or n for no. A fairly crude pictorial representation of the layout will be shown on screen. See Figure 5.
9. Finally, printing takes place.

If you now wish to import the file into Quill to add any bold, underlining etc, do so. Remember you may need to go into the Design menu to set top and bottom margins, lines per page and so on. The BASIC program will have added form feed characters at the end of each page, which are translated into hard line breaks by Quill, which is useful to test that your input and output layouts agree!

That's really all there is to it. As your project until the next issue comes along, why not try to think of areas for improvement.

Add some error trapping.

Add checks to make sure that the number of columns times the column width will fit into the page width specified.

Figure 4 - Entries requested by the program

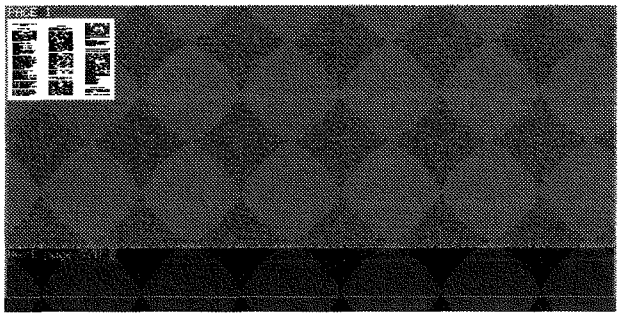


Figure 5 - Previewing the output on the screen

Modify the program to set the appropriate print pitch. For example, there are some REMs around line 400 showing the Epson printer control codes ot use to set the various print pitches.

At the moment, if you don't like the preview output, it goes on to print, unless you BREAK out of the program when it asks you to press ENTER at the end of the preview.

It cannot handle embedded control codes at the moment, without affecting the column layout - lines will be offset left or right and so on. Think of a way of looking at the string it fetches from the source file into the variable txt$ (INPUT #3,txt$) and scan that line for common control codes such as those for bold, underline, etc and alter the number of spaces added in the line after INPUT #3,txt$ to allow for the control code, so that the column edges are printed in the right place.

An alternative to adding spaces to pad lines out to the column start points, especially if using Quill, would be to use TAB stops. This would obviously depend on the TAB handling of the editor or word processor in question, and printers also vary greatly in their handling of horizontal tab stops. It may also allow you to make limited use of proportional printing by defining TAB stops on the printer using the ESC D n1 ... n32 0 command on an Epson printer to define up to 32 tab stops (n1 to n32), one at the start character number for each column, then this program could be modified to print a tab, print the line of text for column 1, print another CHR$ 9 TAB character to move the print head over to the start of column 2, and so on.

The program gets confused if the input text file contains carriage returns (CHR$ 13), since it's a control code and upsets the line lengths and column positions. Try adding a little routine after the INPUT #3,txt$ statement to see if the first or last character in the line is a CHR$(13) and if so strip this off:

```
INPUT #3,txt$
IF txt$(1) = CHR$(13)
 THEN txt$ = txt$(2 TO
 LEN(txt$))
IF txt$(LEN(txt$)) =
 CHR$(13) THEN txt$ =
 txt$(1 TO LEN(txt$)-1)
```

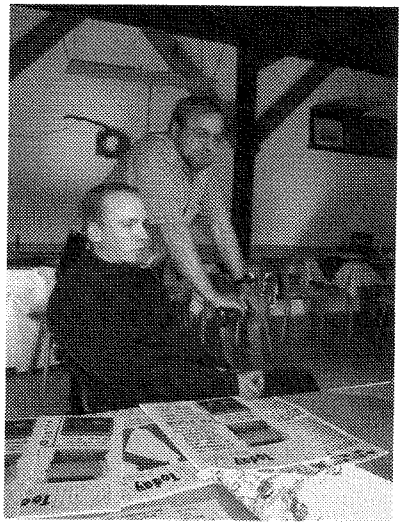
For example, a 2 column layout with lines of 30 characters across each column might look like this:
Left margin: 5 spaces Column 1 : 30 characters across Gap between column 1 and column 2: 10 spaces Column 2 : 30 characters across
This amounts to 75 characters, leaving a right margin of 5 spaces.

Have fun modifying this program. It was originally created for my own use and still used to this day!

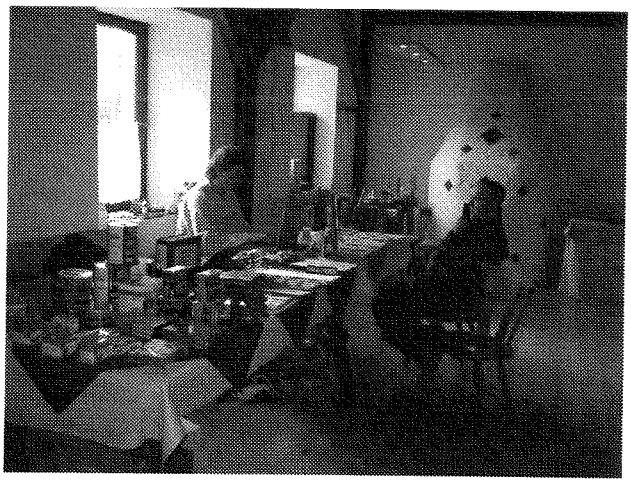
The Berchtesgaden "QL Show" 2004

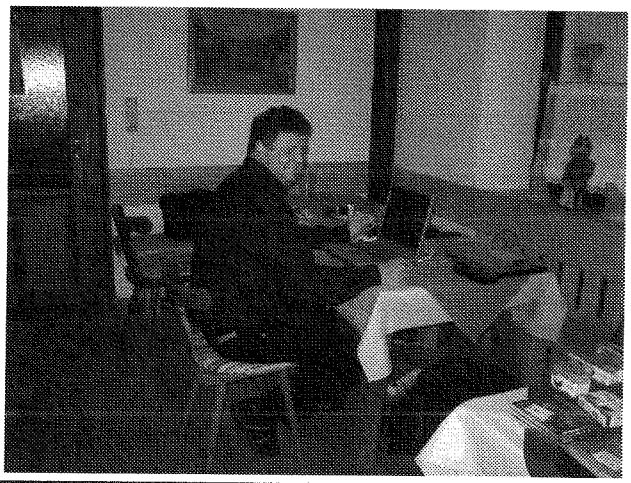
Jochen Merz

For the fourth time, Friedemann Oertel has organised a QL Show in Berchtesgaden.
The first show, four years ago in the same venue (Hotel Schwabenwirt opposite to the Railway station) was very well visited.
Year after year fewer visitors came to the show, but it was always a very nice meeting, which was always followed by a nice meal, long discussions and amusing stories - very entertaining.
The visitors from the UK always came in by plane, a private "Peter Fox" flight. Unfortunately, this had to be cancelled last-minue due to problems with Peters plane.



But even if Peter, Roy and Tony had had made it, it would still have been an empty show. The pictures shows all visitors and vendors, whereby Marcel and Friedemann are shown on two photos.
Still a productive event as several problems of the visitors were solved. It has also led to another improvement in QPC (I am currently beta-testing).
As I used to be in Bavaria at this time of the year (my girlfriend Andrea and I usually end our holiday with the QL meeting) and plans for the next year are already made, I wonder if Friedemann takes up on the effort after this disappointing event again?
Are **YOU** interested in another meeting? Will **YOU** come? If so, please tell Friedemann!
Thank you, Friedemann, and if there is no positive Feedback, we're still happy to come along for a visit (but without carrying all the goods).





QL Software Sources - Part 1

Dilwyn Jones

More than twenty years of Qling later, I thought it high time I published a list of where you can get QL software these days. Of course, you can read the adverts in QL Today and support our traders by buying their commercial software, but there is also a massive amount of free software or shareware out there, and John Perry has described elsewhere in this issue the various types of softwares out there (*wares as he calls it), and it was John's article which inspired me to write this article.

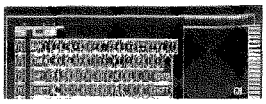
QUANTA LIBRARY

The user group Quanta maintains a massive free software library for its members. If your QL system can cope with QL format CDs (the QXL.WIN type) you can get the whole lot in one go on a single CD-R, thanks to the work of Darren Branagh who put it all together. Or, you can contact the Quanta librarians to get copies of individual floppy disks as required to save having to plough through an entire CD to find what you want. Quanta has a catalogue disk with a menu driven and searchable database of the available programs, making it quite easy to find what you want. While there is some duplication between Quanta library and free software available elsewhere, much of the Quanta software library is unique and available only to members. Indeed, it is well worth joining Quanta just to get access to the software library, never mind the newsletter, helpline and QL shows that they provide (end of free advert).

PD LIBRARIES

The QL has been quite well supported by PD libraries over the years, from people like Richard Alexander of CGH Services in the early days, then Ron Dunnett of Qubbesoft P/D and Steve Johnson of SJPD. There has also been something called the International Freeware Exchange which I was not aware of for many years, and more recently PD library services have been offered by Phil Jordan of The Library, and I have also made free QL software library available on disk as well as through my websites.

QUANTA



Independent QL Users Group

World-wide Membership is by subscription only,
offering the following benefits:
Bimonthly Newsletter – up to 40 pages
Massive Software Library – All Free !
Free Helpline and Workshops
Regional Sub-Groups. One near you?
Advice on Software and Hardware problems
Subscription just £14 for UK members
Overseas subscription £17

Barclaycard: Visa: Access: MasterCard

Now in our Twenty First Year

Further details from the Membership Secretary

John Gilpin, 181, Urmston Lane
Stretford, Manchester, M32 9EH
Tel. +44 (0) 161 865 2872
or
Visit the Quanta Web Site
http://www.quanta.uni.cc
E-mail: quanta_membership @ uk2. net

More recently, the availability of the internet and large capacity storage media such as Zip disks and CD-Rs have more or less made the traditional floppy disk PD library services obsolete and the primary source of free QL software is now the internet or World Wide Web. Another source of free QL software is the various QL Bulletin Board Services, such as the one run by Tony Firshman of TF Services. Others have been run by people like Derek Stewart, Phil Borman, Ron Dunnett and Jochen Merz, using the BBS software made available over the years by people like Jan Bredenbeek and Phil Borman. Many of the BBS'es have fallen by the wayside lately, as the increasing popularity and accessibility of the World Wide Web has meant that fewer people now use the bulletin board systems. Whether you choose to access a bulletin board system or a website to obtain your QL software, one thing you'll find they all have in common is that the files you download will probably be in .zip or _zip format. This means that they have been compressed and packed into a single file called a ZIP file. ZIP is a file compression and archiving program by the Info-zip group. The QL version of ZIP (and UNZIP the program which unpacks the compressed files back into their original form) is maintained by Jonathan Hudson and the programs, complete with sources if you'd like to look at those, are available from his website at **www.daria.co.uk**

Some Software Download Sites

Probably the best known and most highly respected QL software download site is that operated by Thierry Godefroy. It was one of the original QL sites and it has a huge range of software you can download. Figure 1 contains a screen dump of this site and you can see from reading this that it contains quite a large number of categories of software, and each category has a good number of programs to download. The pages also tell you what size the download will be, so that those on fairly slow dial up connections like me can gauge how long it might to download a program. Once upon a time, QL programs were quite compact. Now that we have large memory systems and hard disks, some packages are becoming megabyte monsters like you find on other computer systems, so it's handy having the file size shown in case you get trapped into a long download session which you'd rather do at a time when telephone connections are cheaper, for example. Thierry Godefroy's download site is at:

http://thgodef.nerim.net/smsq/

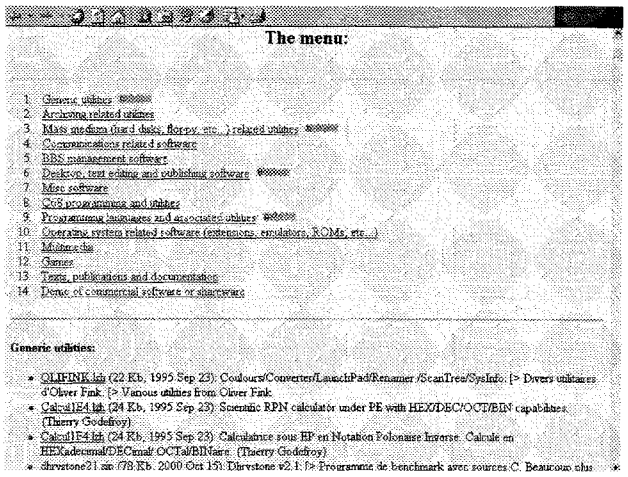


Figure 1 - Thierry Godefroy's QL Software Repository

Another well known and highly respected software site is that operated by Jonathan Hudson. His site contains mostly his own software. He is a prolific software author who has ported vast amount of code from other platforms, using the wealth of software sources out there and recompiled it for QDOS systems, often with Windows and Linux versions as well. Due to the fact that Jonathan includes copious documentation and source files, the downloads from his site may be rather large, but the quality of the software is beyond doubt. Don't let the fact that the site makes extensive mentioning of Linux and Palm Pilot software put you off, there is a wealth of software which runs on QL systems, or on other platforms with the QL in mind, such as the Qltools, QXLTool and WXQT2 programs for transferring files between QDOS systems and other platforms such as Linux, Windows and so on. Jonathan's software downloads are on his wife Daria's site at:

www.daria.co.uk

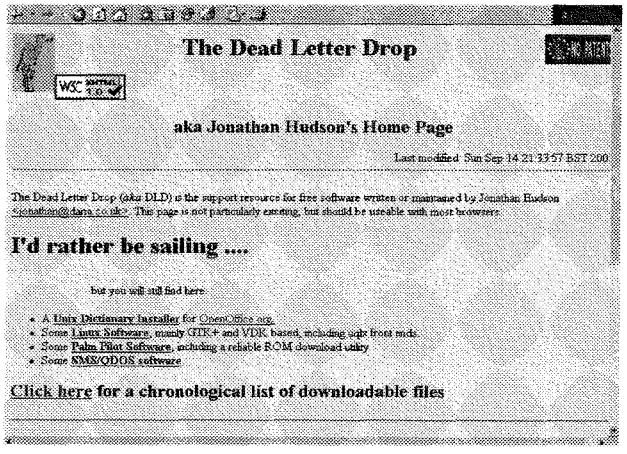


Figure 2 - Jonathan Hudson's 'Dead Letter Drop' site

If you are a Q40 or Q60 user, you'll find a number of applications designed with the Qx0 computers in mind on Claus Graf's website at

www.q40.de

Claus is the brother of Q40 and Q60 designer Peter Graf and the site is devoted to these computers. Click on the Download link to get to the software download page, from here you can download programs to use the graphics and sound capabilities of these computers, as well as general utilities and documentation files.

Figure 3 - The Q40.de site

Proving that the QL is an international system, it has websites hosted all over the world. Davide Santachiara used to trade under the name of Ergon Development in Italy. In recent years, he has released many of his formerly commercial programs as freeware via his website. The site may be viewed in English or Italian simply by clicking on the appropriate flag at his home page, at:

http://www.geocities.com/SiliconValley/Park/6533/

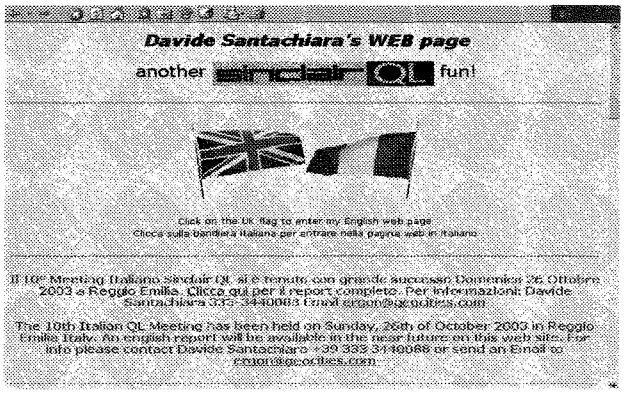
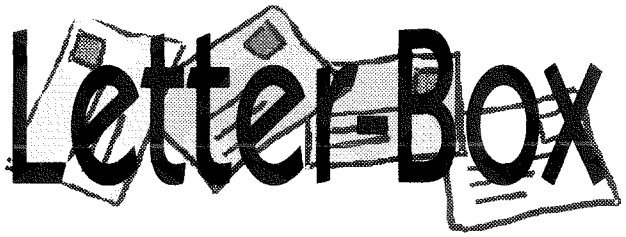


Figure 4 - Davide Santachiara's website



Simon N Goodwin writes:

Dear Editor,

I feel obliged to respond to Roy Wood's continued tizzy about Epson printers.

The first point, which I made here in reference to still-current Brother printers, but which seemed to pass Roy by, is that the ancient ESC/P standard he wants printers to use is still supported by other printer manufacturers, so focusing on Epsons own dwindling and expensive options is misleading himself and – more importantly – readers of QL Today.

The second point is that ESC/P is not the only printer protocol with good support from Qdos. Hewlett Packard's PCL (Printer Control Language) was designed to address the limitations of early printer control sequences like ESC/P, such as their fixed codes for specific resolutions tied to the hardware of 1980 matrix printers like the MX80 (I've still got one of those, but it's out in one of my sheds).

PCL arrived with high resolution square pixel printers like the Deskjet and remains well supported – not least because unlike ESC/P it is scaleable (the magic numbers in the printer control sequences are written as plain text and can use as many digits, and hence as much resolution, as you choose to ask of the printer, with no need for changes to the protocol).

Dilwyn knows ESC/P well from his own print utility coding, and readers of my DIY Toolkit column may remember a colour QL World cover of a Kingfisher printed with PCL code in the mag. The free DIY Toolkit contains screen print and Psion setup commands for any PCL printer. PCL is supported by Quill, Abacus, Archive, Easel, Text87 and many other staple QL apps. Adding support for others is easier than it is for any other protocol I know, as the format is stable and well documented and sample code in BASIC and 68K assembler is plentiful.

Furthermore, as I noted last time, my Brother HL-1050 printer, among others, recognises both ESC/P within the limits of that protocol and – more capably – PCL commands, automatically selecting the correct emulation depending on the first escape sequence it gets from a given port (parallel or USB, for original home computers and emulations respectively).

All this fuss, and the recommendation that we have little choice but to pay over £200 for a QL-friendly printer, ought to be challenged.

That said, for anyone with a 68040 or 68060 the idea of using Postscript (via Ghostscript) as an Esperanto is a fine one. However it is too hungry for memory and CPU time to be usable on 68000 systems, and inefficient and notably slower than current printers even on 68020/030 setups.

Ghostscript can write PCL directly, by the way. So if you opt for a PCL printer, you've got both options covered. :-)

Back in the distant eighties when I was first using my QL I used to wait eagerly for the next edition of QL World so I could see what new programs had been released and what new uses people had put their QLs to. In that time there were new additions on a regular basis and trade was fairly brisk, although, for my part, it was limited by my lack of spending power. As we now move into the QL's old age the lack of new programs and hardware is taking its toll on the enthusiasm of the users. Many of our users plough their own furrow writing esoteric code to do things that they alone would want to do and many more continue to use the same batch of programs they have always used.

'What's wrong with that?' you many say and I suppose that there is little answer to that except to say that the scene was healthier in those bygone days because of the activity it provoked. It was in some sadness, therefore, that I noted Thierry Godefroy saying that he had ceased to be a QL developer although he is still actively using his QDOS/SMSQ systems. I won't go into any details here but those of you who frequent the user group lists will know the circumstances.

Many other programmers have fallen by the wayside over the years leaving us with a scarcity of new programs. There are also precious few updates and improvements to existing programs and fewer people porting from other systems. We do need to revive the situation somehow and persuade some of our existing programmers to be a little more productive and maybe even adventurous.

There are a whole host of uses a QL could be put to and several changes that could be made to existing systems which would benefit the current batch of users.

Stuff it!

One thing I have been nagging on to people about for a while is a utility that I use a lot on my PC systems. The Windows system has a very rudimentary clipboard and can only store one item. As each one is copied the previous one is displaced. I have a small shareware utility which will store all of the data copied to the clipboard and which can be called up with a simple keypress to re-insert any of the items stored. This list is preserved even when the system is shut down and re-started.

The QL's Stuffer Buffer has a history already but the items can only be recalled by a series of keypresses in the reverse order to that in which they were copied. If we had a utility which would capture and save a list of items sent to the stuffer buffer and then allow us to select any of these from a list then it would be much better. Jochen Merz's programs have long had the scrap facility and that could be incorporated into this utility too allowing items to be passed from Scrap to Stuffer Buffer and vice versa. Something like this would be a worthwhile piece of software and if there is someone out there who would like to write it I would be happy to market it.

Sending out an SMS

There is also the case of the Short Messaging Service as used on mobile phones. I did use a utility on the laptop which would allow me to send messages to phones via a modem. This was a fairly unsophisticated piece of software which only required a knowledge of the format used and the numbers to dial. This is, again, something which would be of use to many people although the downside is that it cannot receive replies.

The Big Divide

All of the above leaves us in the situation whereby the QDOS/SMSQ system is less of a tool towards a productive end and more of an icon or cause celebre amongst the users. Those who still use the system as a working environment are being more and more marginalised by the people who have large axes which need sharpening. By this I mean those who trumpet the freeware licence above the actual use that the software is being put to. This is counter productive in the extreme and serves only to drive away the actual end users of the system.

If that were not enough the prolonged arguments about licensing and other factors have driven the people who actually use the system for what it is to abandon the user group list and boycott it's publications.

I know we have aired these differences and arguments before but there does need to be

a healing of the wounds and a return to the spirit of co-operation that was always the hallmark of the QL community. If we could re-kindle the spirit that existed before we might get some useful progress.

Patently Oblivious

Software patents have been hitting the news in computer programming circles and this is something we, in the QDOS/SMSQ community should be aware of too. Moving away from the discussion of licensing we come up against a brick wall of quite monolithic proportion.

Many of the larger software companies in the US (and I do not mean just the reviled Micro$oft here) have pushed the legislature into passing stringent patent laws and declaring the ideas and concepts used to create original programs to be 'intellectual property'. This is a frightening prospect for the many mainstream software companies because it does mean that they cannot use many of the accepted methods of creating programs for fear someone has patented part of the procedure and thus they would be left having to pay a royalty.

Now I am all for the protection of people's work and I have long said that the many people who use illegal copies of software are indulging in a form of theft. There are many who would argue about this and some get quite upset when I mention it to them. One of the people I do some PC work for was very shocked when I would not give him a copy of Microsoft Office and even more shocked when I told him that the software on my machine was all legal and registered. they invoke elaborate argument along the lines of 'The software is all rubbish and bugged anyway so why should I have to pay for it?' Well why should they want to use it?

They also say that M$ has so much money they don't need their contribution and would not miss the money but then, compared to the homeless man sleeping on the street I would imagine that they seem like they have too much money and wouldn't miss some of their possessions either.

I don't want to preach here, I have used pirated software myself in the past but it is this wholesale theft of software that has led it the annoying way that software now has to be 'activated' and led many of the companies to more and more elaborate ways of protecting their property. It has also detracted from the amount of time the programmers spend on actually writing the software.

These practices have led to the 'Intellectual Rights' laws in the US and these laws are being considered for ratification by the EU. In essence they mean that company X can declare a patent on the idea that you store all your records in card index style database. If they had written one a while ago or can show work which led to the development of this idea they would get the right to sue anyone else using a card index style database and claim royalties.

Vampires of the Mind

This is all going far beyond the accepted uses of patent law to prevent large companies from stealing the ideas of individuals or smaller companies and, in many ways can be so restrictive and stifling that no one will gain from it.

You may well say that no one would come after a QL programmer if they wrote something which infringed these laws but that is beside the point. Many of our programmers in the past have not added features to their programs or products because they infringed EU law / Licences / etc. Stuart Honeyball cited the introduction of the laws on Electronic radiation and the requirement that equipment sold within the EU has a CF certificate as one of the reasons for his withdrawal from QL hardware development and PROGS were not able to include GIF production tools in their software because of the savagely aggressive attitudes of Compuserve who owned the licence. I am sure that many other QL programmers and developers will be just as moral in their attitudes.

In a sense this kind of legislature is draining the lifeblood of ideas from peoples minds and should be resisted at all costs. I do not know how far the laws have got in the EU parliament but we should make an effort to thwart the behemoth of law in any way possible. (Dilwyn, will not be able to help using this because one part the EU machine recently produced a map of the whole of the EU - leaving Wales out. Does this mean he has been ejected?)

Just Browsing

A lot was said about PC browsers in the QL lists this month and some of the users were very vocal about security and the Internet. Un-surprisingly, many of those who were very

down on M$'s Internet Exploder were very silent about the revelation in the last week that one of the browsers they put forward had a security hole in it that you could sail the Titanic into. Without going too deeply into the issue it seems that Mozilla had to issue a hurried update because users of it's Firefox browser were vulnerable to be hacked into and, worryingly have the contents of their drives deleted or altered. Nice.

I have mentioned before that the reason that M$ seem to have so many security problems is that there are so many people out there who want to either exploit the bugs or just jab the company sharply in the ribs with a pointed stick. I fully expect that, if more people use the alternate browsers more holes will be found and more problems emerge. As code gets more complex it becomes harder and harder to maintain an overall view of what is happening. Add to that the fact that every time you patch a piece of code you change part of it from how it was originally conceived and could be introducing other errors and problems somewhere else. As I said when we last discussed M$ both on the user group and in person at the US show. I am no apologist for M$ but it does not pay to be too smug about your alternative choice.

It's a RAID

I did not need a malformed browser to destroy my system in the last weeks before getting this article ready for the printer. I turned the machine on and got an immediate blue screen informing me I had problems with the Windows installation. PCs have got to be so complex these days that straightforward troubleshooting is not straightforward at all. After a bit of probing and testing I came to the conclusion that the motherboard was not seeing the PCI slots correctly and, since the main boot hard drive was a raid array on a PCI card I thought that had to be the problem. I could boot from a single drive plugged into the motherboard's IDE slot and the motherboard itself had been giving the odd error so I decided a new board was in order. Of course this also meant a new CPU and RAM because the RAM and CPU I had in the old system were out of date and I could not get a good motherboard which would support them.

To be doubly sure I also replaced the Power Supply which had been a little noisy. After a rebuild I managed to get the system up and running on a single drive and installed all of the drivers and software for the new board. By the next Thursday night I was fairly confident I had got the thing under control so I tried to put the raid array back together. This immediately fell over. When I plugged the single hard drive back in the system started to boot only to stop with one of those wonderful error messages that Windoze does so well. 'You either have no pagefile or you page file is too small. To correct this go to My Computer and.....' Trouble was, of course, it would not boot far enough to get to the 'My Computer' screen. A few hours of head scratching later and, after I had transferred the drive to an external USB box, plugged it into my laptop and tried all the ways I knew to get the pagefile (that area of the hard drive used by Windoze as virtual memory) back, I gave up.

I decided to format one of the drives and re-install from scratch. After the re-install I backed up the drive to another one and tried to rebuild the raid array again. Without success. Its seems it was the raid card itself which fell over. Oh Well.

All in all it took me over a week (evenings only because I work in the day time) to get my system back together. The one satisfying part of the exercise was that fact that all I had to do was to copy the QXL.WIN file back onto the newly formatted Hard Drive and copy the QPC2 folder there too and I had my QL system running in seconds. In these days of superfast highly complex PCs you can do an awful lot of things but for sheer rugged, survive-it-all, efficiency you cannot beat a simple old QL system.

My PC was roughly two years old. All of the components were outdated and obsolete when I wanted to replace them and, because I went for a board with an AGP slot to avoid buying a new graphics card, the Graphics will be out of date by next year. On the other hand my MinisQL is still going strong with the same motherboard, CPU and RAM as when I bought the Super Gold Card from Miracle in Eindhoven in the early 90's and the Aurora from Qubbesoft in the first month of its launch. to return to the start of this column I still think it is a pity that, given the fact that we have the best and most highly maintained O/S we have had for the QL in SMSQ/E for some years, it is a great pity that more people are not using for it and trying to use its facilities.

All Aboard

During the late summer I got the chance to go with Marcel and Tony to see the boat that veteran QLer Jens Wildgruber had been sailing around the UK. He had sailed into Brighton and Marcel had flown over to the UK so he could join him sailing back to Germany. Jens invited us aboard for the evening and Tony came over just to have a look at the boat and see what use Jens had put the QL hardware he had bought from him to. He had a full Aurora/SuperGold Card setup built into the boat. Now that is an example to all of you about innovation! Thanks for the wine and hospitality.

Unpowered Users

Finally, back to another little hobby horse of mine. One of our long time users revived his QXL card on an old PC and wanted to move files between the DOS and the QL sides of the machine. He asked for help and many people popped up to offer help. He had to struggle with the various problems of using QXL tools driven by the command line (even worse for him was the first version of this he found on the net which required compilation). This was a non technical user and really there should be versions of this software with front ends included so it can be used. I realise that writing these interfaces is not the most exciting part of the program and that many of the more 'power user' people out there regard front ends, particularly graphical ones, as the digital equivalent of frozen omelettes but this is the sort of thing that the PC user can take for granted and we are the weaker for not having it.

Of course he could have bought QPC2 and just done it with ease from there. Is there any way to get the DOS devices into the QXL SMSQ/E?

I realise that the QXL is a very old piece of technology and it is not a fast as QPC2 but it is also a very good way of using an old PC. All current boards do not have ISA slots so, if you have an old machine around this is one way to get to do something useful. Having QPC2's DOS devices in the QXLs SMSQ/E would be a very useful add on. If they a commercial addition to QPC2 maybe can they be supplied as a cheap stand alone option?

The German Show

Tony and I were unable to attend the German QL show at the last minute because Peter Fox, who had volunteered to fly us there, had problems with his plane and it was too late to book a cheap flight. This was a great shame and I hope that we will be able to attend another show there later. He had bought new engines for it and although they had arrived the servo injectors had not and the plane was grounded. I told him at the time that, much as I wanted to be able to go I was not going to push the plane all the way.

Finally a word about this column. I realise that a larger part of this issue's Byts of Wood is taken up with more general computing matters. Some of this is because it was mentioned in the users list but it does have a bearing on the amount of activity in the QL scene. I hope that many of you will be reading this at QL 2004 in Eindhoven and next year the QL will be 21 years old. Given that my motherboard and CPU were obsolete when I came to replace them after only 2 years service it is remarkable that so many of you are still working with 21 year old computers. Maybe QL 2004 will inspire a little more activity in the

The next issue of QL Today will probably be released in the middle of December, and hopefully reach the European readers just before Christmas.

Due to the shift of the release for QL 2004, we think we should stick to the release date in the middle of even month until another main even requires another shift forwards or backwards.

We are flexible, but remember, the magazine also depends on YOU and your articles...

The QL Show Agenda

QL Meeting - (UK) Byfleet

Sunday, 28th of November, 10:00 to 16:00

Byfleet Village Hall

Same venue as all the years before!

The Hall is just inside M25, between jns 10 & 11, and just South of A245.

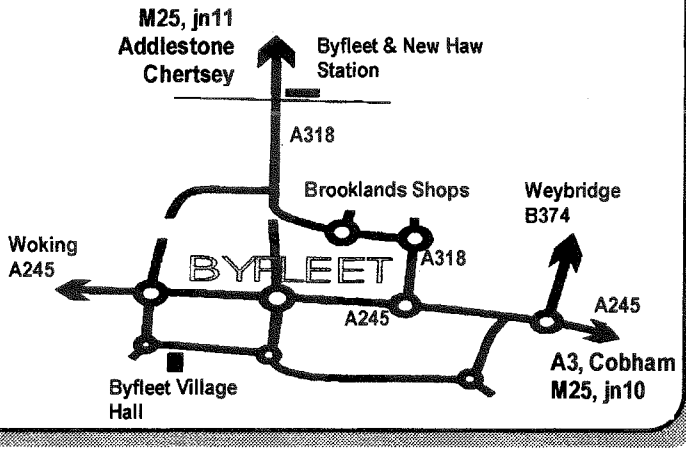
›From M25, jn11; go East towards Weybridge, then turn right (South) onto A318 and follow it to Brooklands (now a business park), then through the old racetrack, turning right onto A245,. Turn left at second roundabout, left at little roundabout, and Hall is on right.

›From M25, jn10; take A3 towards London, left onto A245 towards Woking at next junction, (Painshill). After A318 joins, turn left at second roundabout – see below. (Or you can take the earlier left fork into Byfleet, go right at first small roundabout and left at the next, by the green.)

›From A3, either direction; leave at Painshill junction with A245 Cobham/Woking head for Woking.

By train; Byfleet & New Haw is on the Waterloo Woking line. It s 12 to 15 mins walk. Don t go through Brooklands – carry on down the old road, across the A245, and on to the green, then turn right.

Free Parking, and all the usual attractions. If that isn't enough, Brooklands Museum (aircraft and motor racing) is just up the B374: or there's a Bus Collection on the A245 to Cobham: or I'll explain how to get to the RHS Gardens at Wisley (RHS Membership needed on Sundays).



Future QL Shows

We hope that QL 2004 will be a successful meeting, and we hope it will encourage users to visit more QL shows.

The US QL show had very few visitors, Berchtesgaden was extremely disappointing (see report inside).

Do you want any more QL meetings?

Do they serve any useful purpose or should we stop them?

Do you want one major yearly event? If so, where? Which time suits you best?

Is this lack of attendance purely the result of a lack of new products?

As travelling costs are increasing every year, and sales revenue is decreasing, we, the QL "dealers", more or less sponsor every show.

If the shows are no longer of interest to the people who actively use QDOS/SMSQ software then there is very little point in continuing to hold them. If we do decide to stop holding shows, however, we will have little direct contact with our customer and user base. This is something we would like to avoid if possible.

We need feedback from our customers. Contact us through any of the QL Today addresses, by snail mail, email or fax and we will publish the results in the next issue and use them to decide next year's meeting schedule.